
Title: Context-aware gestural interaction in the smart environments of the ubiquitous computing era

Name: Maurizio Caon

# CONTEXT-AWARE GESTURAL INTERACTION IN THE SMART ENVIRONMENTS OF THE UBIQUITOUS COMPUTING ERA

by

MAURIZIO CAON

A thesis submitted to the University of Bedfordshire in partial fulfilment of the requirements for the degree of Doctor of Philosophy

June 2014

CONTEXT-AWARE GESTURAL INTERACTION IN THE SMART ENVIRONMENTS OF THE UBIQUITOUS COMPUTING ERA

MAURIZIO CAON

ABSTRACT

Technology is becoming pervasive and the current interfaces are not adequate for the interaction with the smart environments of the ubiquitous computing era. Recently, researchers have started to address this issue introducing the concept of natural user interface, which is mainly based on gestural interactions. Many issues are still open in this emerging domain and, in particular, there is a lack of common guidelines for coherent implementation of gestural interfaces.

This research investigates gestural interactions between humans and smart environments. It proposes a novel framework for the high-level organization of the context information. The framework is conceived to provide the support for a novel approach using functional gestures to reduce the gesture ambiguity and the number of gestures in taxonomies and improve the usability.

In order to validate this framework, a proof-of-concept has been developed. A prototype has been developed by implementing a novel method for the view-invariant recognition of deictic and dynamic gestures. Tests have been conducted to assess the gesture recognition accuracy and the usability of the interfaces developed following the proposed framework. The results show that the method provides optimal gesture recognition from very different view-points whilst the usability tests have yielded high scores.

Further investigation on the context information has been performed tackling the problem of user status. It is intended as human activity and a technique

based on an innovative application of electromyography is proposed. The tests show that the proposed technique has achieved good activity recognition accuracy.

The context is treated also as system status. In ubiquitous computing, the system can adopt different paradigms: wearable, environmental and pervasive. A novel paradigm, called synergistic paradigm, is presented combining the advantages of the wearable and environmental paradigms. Moreover, it augments the interaction possibilities of the user and ensures better gesture recognition accuracy than with the other paradigms.

DECLARATION

I declare that this thesis is my own unaided work. It is being submitted for the degree of

Doctor of Philosophy at the University of Bedfordshire.

It has not been submitted before for any degree or examination in any other University.

Name of candidate: Maurizio Caon Signature:

Date: 26 June 2014

# List of Contents

# Chapter 1: Introduction

## 1.1 Background and Motivation

Three important waves have been identified in the relationship between human beings and computers (Weiser & Brown, 1997). The first wave has been called “Mainframe”, when many people shared the same computer. The second wave came with the “Personal Computer”: every person has her/his own computer. The “Ubiquitous Computing” era indicates the third wave, where many inter-connected computers share each user (Figure 1.1). In this era, the computers must be pushed into the background of the everyday life, transparently integrated in the physical world granting a seamless interaction with the digital information. Weiser referred to this concept as calm technology, where people are aware of the augmentation of the physical world by the embedded computation capabilities but computers do not constitute a barrier to personal interaction.

In this era, the interaction between humans and computers should be more natural. The raising awareness about the importance of putting the human being and his needs at the center of the system design obviously led to a revolution of the human-computer interfaces. This change brought the understanding that the user should not be forced to sit in front of a single glowing screen while pushing an array of buttons. In fact, the natural way of interaction among human beings is primarily based on speech and gestures. Therefore, in order to make Human-Computer Interaction (HCI) more natural, a system should get closer to these forms of multimodal communication (Krahnstoever et al., 2002). In particular, gestures represent a powerful means of interaction since they allow convoying messages through expressive movements for the communication with other

human beings and they also allow the manipulation of objects for the exploration of the surrounding environment. Nowadays, gestural interfaces represent a main trend in the HCI domain. In fact, many research works can be found in the literature about gestural interfaces. However, these works are often specific solutions and this led to the major issue in the gestural interaction domain: a lack of common guidelines that can enable designers and engineers to create intuitive interfaces (Norman & Nielsen, 2010). One main concern consists of the ambiguity of gestures; indeed, the same gesture can have several different meanings depending on the context, which sometimes makes its decoding hard even for humans. This issue is also due to the heterogeneous nature of gestures and that compelled the psychologists to introduce some classification methods for the gesture categorization. These gesture categories were eventually adopted in the HCI domain to provide a basic principle for the design of gestures. Unfortunately, the gesture classification does not provide any guideline for the design of gestural interfaces. In particular, there is a lack of a high-level framework that could enable the design and development of context-aware interfaces for the human-environment interaction based on gestures. This framework should comprehend every type of gesture for the interaction with smart environments and should also provide a systematic approach for the gesture design in order to make interfaces more intuitive, hence, more natural. The research conducted in this thesis deals with this issue in the frame of ubiquitous computing, following the principles of Weiser's calm technology.

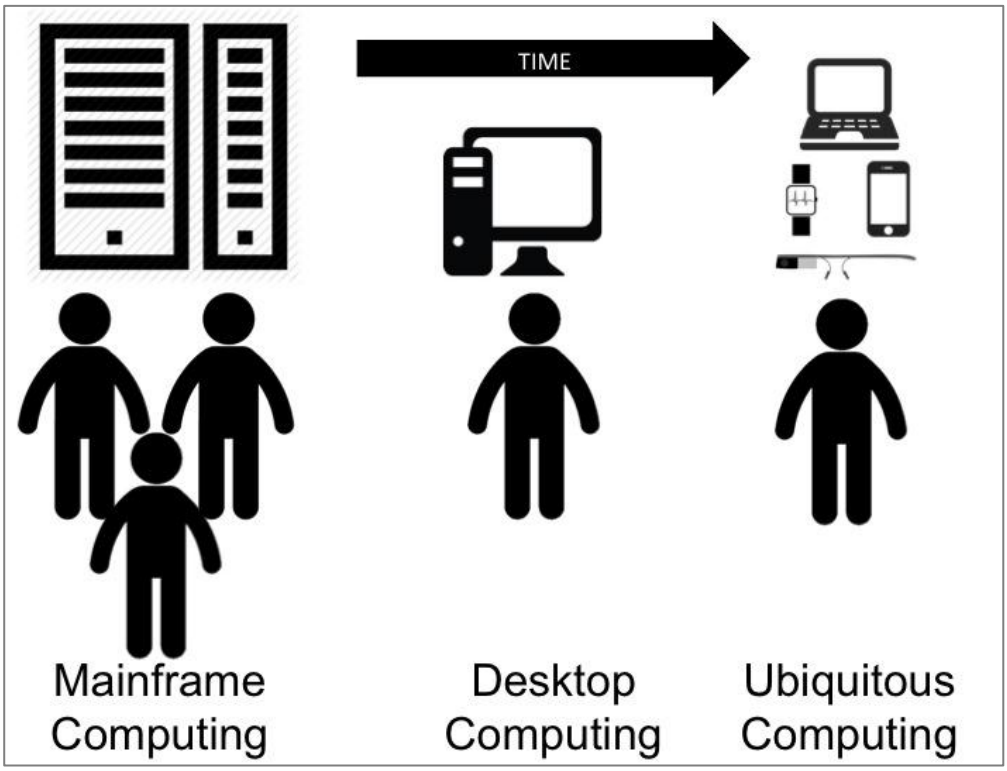


**Figure 1.1 The three waves of human-computer interaction.**

## 1.2 Research aim and objectives

This research aims to analyze and devise a framework for context-aware gestural interaction in the smart environments of the ubiquitous computing era, proposing a new approach for gestural interface design based on the aforementioned framework in order to improve the user experience.

The specific objectives of this research are to:

- Investigate and analyze the current approaches for gestural interaction in smart environments highlighting the different paradigms of ubiquitous computing.
- Design and implement new algorithms to improve gesture recognition in smart environments.

- Test and verify the correctness of proposed algorithms based on tests for the gesture recognition accuracy.
- Devise different scenarios of application of this research and develop different prototypes following a user-centered design approach. In particular, the prototypes developed will be implemented following the proposed framework and gesture design approach and used for tests with real users in order to evaluate the user experience with particular emphasis on the system usability.
- Explore the possibility of expanding the context information introducing the concept of user status, which is mainly based on human activity recognition, with reference to the ubiquitous computing paradigms.
- Develop a prototype adopting a novel approach and test the recognition accuracy.
- Introduce the concept of system status as important factor for context-aware gesture recognition with reference to the pervasive computing paradigm of the ubiquitous computing.
- Develop a prototype to assess the improvement concerning the gesture recognition accuracy and the user experience.

## 1.3 Research methodology

The research methodology central to this work is as follows:

1. Literature review: Review and summarize previous researchers' works in relation to this work.
2. Problem definition and analysis: present existing approaches for gestural interface design; review existing interfaces for gestural interaction in the smart environments with reference to the ubiquitous computing paradigms and to the current techniques for view-invariant gesture recognition; study the literature concerning the context-aware gestural

interaction. The problem of the research is defined with proposed aims and objectives.

3. Prototype system design and implementation: Design and implement a prototype system to test the proposed gesture recognition technique and the proposed gesture design approach based on the new framework. Design and implement a prototype system for the human activity recognition to integrate the context information with the user status. Design and implement a prototype to integrate the context information with the system status in order to improve the user experience. The design and implementation of these prototypes should follow the user-centered design process.
4. Experiments: Three experiments have been conducted. The first consisted of two phases: first, the prototype system testing and evaluation to assess the gesture recognition accuracy of the proposed algorithm; second, the same prototype has been applied to different scenarios in order to evaluate the user experience with particular emphasis on the usability. A second experiment aimed at evaluating the activity recognition to integrate the context information. The third experiment involved the test of the prototype for the system status.

## 1.4 Overview of the thesis

Chapter 2 reviews the literature work conducted on gestural interfaces for the smart environments of the ubiquitous computing era. It starts describing how gestures evolved as communication means in humans, what is its role in the current human communication and the classification system in psychology. Then, the gesture classification system for HCI is presented and the related problem of the current gestural interface design is explained with reference to the proposed approach and the existing techniques. Afterwards, the different paradigms of the ubiquitous computing are presented and explained with reference to the

gestural interfaces of smart environments extracted form the literature. The end of this chapter is dedicated to explain why it is important to implement context-aware gesture recognition and the current proposed frameworks for the user interface layer are presented.

Chapter 3 presents a novel high-level framework for context-aware gestural interaction with smart environments. Moreover, a new systematic approach for the design of functional gestures is presented to enable practitioners to design optimized gesture taxonomies for an improved user experience.

Chapter 4 describes the application of the high-level framework for context-aware gesture recognition to the development of a prototype as proof-of-concept. The development of this prototype provided also the occasion to introduce a novel technique incorporating contemporary methods and technologies for the view-invariant 3D gesture recognition.

Chapter 5 analyzes the user status as part of the context information as presented in the framework. In this chapter, the user status is mainly based on human activity recognition and a novel technique based on electromyographic signals is proposed. Then, the prototype development and testing are described in detail.

Chapter 6 analyzes the system status in relation to the ubiquitous computing paradigms as part of the contextual information and the synergistic paradigm is introduced. This novel paradigm allows combining the advantages coming from the wearable and environmental subsystems in order to improve the gesture recognition accuracy and the interaction possibilities, which provide a better user experience. The chapter also describes the prototype development and testing.

Chapter 7 is dedicated to the conclusions and the future work.

## 1.5 Thesis related publications

- Maurizio Caon, Julien Tscherrig, Elena Mugellini, Omar Abou Khaled, Yong Yue, " Context -Aware 3D Gesture Interaction Based on Multiple Kinects" , First International Conference on Ambient Computing, Applications, Services and Technologies (AMBIENT 2011).
- Maurizio Caon, Julien Tscherrig, Yong Yue, Omar Abou Khaled and Elena Mugellini, "Extending the Interaction Area for View-Invariant 3D Gesture Recognition", 3rd International Conference on Image Processing Theory, Tools and Applications (IPTA 2012).
- Maurizio Caon, Stefano Carrino, Simon Ruffieux, Elena Mugellini and Omar Abou Khaled "Augmenting Interaction Possibilities Between People with Mobility Impairments and Their Surrounding Environment", 1st International Conference on Advanced Machine Learning Technologies and Applications (AMLTA 2012).
- Francesco Carrino, Stefano Carrino, Maurizio Caon, Leonardo Angelini, Elena Mugellini and Omar Abou Khaled "In-Vehicle Natural Interaction Based on Electromyography", 4th International Conference on Automotive User Interfaces and Interactive Vehicular Applications (AutoUI 2012).
- Maurizio Caon, Yong Yue, Julien Tscherrig, Omar Abou Khaled and Elena Mugellini, "Democratizing 3D Dynamic Gestures Recognition",1st IEEE Workshop on User-Centred Computer Vision, IEEE Winter Vision Meetings 2013. *Best paper award*
- Maurizio Caon, Leonardo Angelini, Yong Yue, Omar Abou Khaled and Elena Mugellini, "Context-Aware Multimodal Sharing of Emotions", 15th International Conference on Human-Computer Interaction (HCII2013).
- Leonardo Angelini, Maurizio Caon, Francesco Carrino, Stefano Carrino, Denis Lalanne, Omar Abou Khaled and Elena Mugellini, "WheelSense: Enabling Tangible Gestures on the Steering Wheel for In-Car Natural

Interaction", 15th International Conference on Human-Computer Interaction (HCII2013).

- Stefano Carrino, Maurizio Caon, Omar Abou Khaled, Rolf Ingold and Elena Mugellini, "Functional Gestures for Human-Environment Interaction", 15th International Conference on Human-Computer Interaction (HCII2013).
- Maurizio Caon, Francesco Carrino, Antonio Ridi, Yong Yue, Omar Abou Khaled and Elena Mugellini, "Kinesiologic Electromyography for Activity Recognition", International Conference on PErvasive Technologies Related to Assistive Environments (PETRA2013).
- Francesco Carrino, Maurizio Caon, Antonio Ridi, Omar Abou Khaled and Elena Mugellini, "Optimization of an Electromyography-Based Activity Recognition System", 4th IEEE Biosignals and Biorobotics conference (ISSNIP).
- Maurizio Caon, Yong Yue, Giuseppe Andreoni and Elena Mugellini "Atelier of Smart Garments and Accessories", in adjunct proceedings of ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp 2013).
- Leonardo Angelini, Francesco Carrino, Stefano Carrino, Maurizio Caon, Denis Lalanne, Omar Abou Khaled, Elena Mugellini “Opportunistic Synergy: a Classifier Fusion Engine for Micro-Gesture Recognition”. 5th International Conference on Automotive User Interfaces and Interactive Vehicular Applications, 2013.
- Maurizio Caon, Leonardo Angelini, Omar Abou Khaled, Denis Lalanne, Yong Yue, Elena Mugellini, "Affective Interaction in Smart Environments", ICT-SB workshop at 5th International Conference on Ambient Systems, Networks and Technologies (ANT-2014), 2014.

# Chapter 2: Literature Review

## 2.1 Gestural interaction

Speech and gestures are considered the main modalities of the human communication. In the current society, speech plays a more important role in the human-to-human communication but often it is accompanied by gestures. Gestures represent a very important communication possibility; in fact, the sign languages performed by deaf people show the full potential of this modality. Signed communication gave the first hint to anthropologists that the oral language comes from the gestural one (Volterra et al., 2005). In fact, one of the most credited theories of the origin of the human language claims that man started talking with hands before the invention of the word. 16 million years ago, the great apes that differentiated from the Old World monkeys had sufficient cognitive skills to develop a protolanguage. Later, 5-6 million years ago, the bipedalism was the main characteristic of the hominids and that allowed them to have free hands while walking. The free hands allowed more effective gesturing and with the growing cognitive skills, they were able to develop a more complex gestural communication form. Only with the emergence of the Homo genus, more or less 2 million years ago, the language started to have a true grammatical structure. From then, the hominids' language kept becoming more and more complex and sophisticated but still mainly composed of gestures. The Homo sapiens started to convey the communication principally in the oral form only 50.000 years ago (Corballis, 2002). Gesture was not simply replaced by speech but gesture and speech together have coevolved in complex interrelationships during the whole human evolution.

Nowadays, gesturing is intrinsically part of the human communication. Some researchers tried to find out why people feel the need to communicate through the movements of hands even while talking. Rimé and Schiaratura investigated if the gestures represent the compensation of a lack associated to the expressivity of the verbal communication, if the motor activity helps the externalization and the sharing of personal representations (Feldman & Rimé, 1991). Their experiments showed that even if the communication subjects could not see each other they kept the same amount of body movement while communicating. This result and other evidences suggest that the gestural motor activity of a speaking person is inextricably linked to his/her verbal encoding activities. Thus, the gestures and body movements are not only an additional communication channel but they represent a sort of embodied thinking for the communicating person.

Rimé presented two classes associated to the gestural forms based on previous studies. These two classes are defined as “speech styles” (Rimé, 1983).

The first speech style is a direct verbal-gestural style; this style is poorly articulated, weakly codified and very subjective. The verbal-gestural style derives from the cognitive-motor view of expression; according to this principle, these depictive gestures cannot be explained without considering the contribution of motor processes to shaping representations of reality. In fact, the motor processes contribute to perception in four different ways. The first way is related to the intrinsic dependence of the organism on the sense organs; in particular, the organs’ activities strongly depend on the motor tracking of stimuli coming from head and oculomotor movements, and motion performed by other bodily parts. Feedback information from these motor processes should be considered as part of the sensory data. In fact, humans have a multitude of senses; the most popular senses are five: sight (ophthalmoception), hearing (audioception), taste (gustaoception), smell (olfacoception or olfacception), and touch (tactioception). The five aforementioned senses are traditionally recognized but the humans’

ability to detect other stimuli beyond those governed by the traditional senses exists, including temperature (thermoception), kinesthetic sense (proprioception), pain (nociception), balance (equilibrioception), and various internal stimuli (e.g. the different chemoreceptors for detecting salt and carbon dioxide concentrations in the blood). Since what constitutes a sense is a matter of some debate and not all these senses can safely be classified as separate senses in and of themselves they are not considered being part of the traditional senses. Nevertheless, they play an important role in the human perception of the reality and memorization of experiences. This concept leads directly to the second relation between motor processes and perception: the person involved in perception and reality processing is usually active, and the sensory responses are continually adapted from moment to moment; hence, the efferent and afferent data are necessarily associated in stored information. Rimé and Schiaratura explained this phenomenon with a brilliant example: “anticipatory head, hand, and leg movements by the baseball player ready to catch the ball, are blended with eye-tracking movements and with the picture of the ball crossing the sky” (Feldman & Rimé, 1991). The third point refers to affective and emotional reactions; in fact, most stimulations coming from the perception of the reality elicit emotional reactions expressed as postural or facial changes. Also in this case, the efferent information is involved in the formation of mental representations. The fourth dependence has been already revealed by Aristotle, who considered that “human being is the most mimetic of all animals, and it is by mimicking that he or she acquires all his or her knowledge”. This quote indicates that human being do not limit the reality representation to selecting verbal attitudes to qualify objects or events; the human beings make also abundant use of motor coding by means of motor attributes. These motor attributes, as it happens for the verbal ones, are numerous for every object and they are hierarchically organized according to their perceived saliency. This phenomenon explains why when a person fails to remember the name of a specific object can still recall and display gesturally the motor codes to represent its attributes.

The correlation between the perception of the reality and the motor processes has been first revealed by Jacobson in 1931 (Jacobson, 1931). He performed several studies on this matter and found out that people, who are thinking of certain events or imagining specific actions, presented a mental activity accompanied by electrical activity in the muscular groups associated with the imagined actions. These first evidences started the following studies to the point of declaring the representation of the reality as constituted of three types of element: 1) concepts and verbal attributes to be articulated into language propositions, 2) images, and 3) incipient somatotonic changes reminiscent of the various motor responses involved in the perception of the referent. The last two elements concur and through their interaction form the active reminiscence of the apprehension process; the verbal attributes are more structured and are both part and consequence of the cognitive process of the other two elements. Each of the three elements has the property to elicit the other two and is a proper input to access to the network of interconnected elements that constitutes the perception of the reality. These elements have also the possibility to be part of other networks of different representations. The difference between the motor-sensory data and the other two types of element is that the first type usually goes unnoticed by the perceiver; instead, the images remain more consciously impressive and the verbal concepts are the outcome of an inner rationalizing speech. The continuous repetition of this acquisition process with the rationalization through the silent cognitive work provides the network of the experience and labels it with concept to from a first raw structure. This first outcome is progressively articulated to make an expressive structure, where the raw data of experience is organized in accordance with the rules of rationality and language and is ready for verbal expression. However, the raw structure is never erased, even once the more elaborated expressive structure has been processed. The expressive structure allows people to overcome the difficulties coming from the task of processing a complex multi-dimensional matrix of information in order to express it as a one-dimensional string of verbal

communication. The expressive structure allows people to verbally express concepts through a more articulated, syntactically correct and fluent speech. Rimé and Schiaratura call this speech style the elaborate or mediated one. Usually, this speech style is highly articulated, distant from the direct experience, thus more objective and abstract. In this case, the speech evolves from very elaborated structures of representations and it is expected to be mainly accompanied by speech-marking hand gestures. The speech-markers are gestures that parallel the formal and rhythmic properties of speech, as the most integrated into the verbal content. Thus, they play only an ancillary role during the verbal expression, being restricted to a self-monitoring and clarifying function. These hand gestures are very different from the depictive ones usually associated to personal experiences and, hence, characterizing the direct verbal-gestural style. In this case, gestures are central in the communication and express a visible expression of the speaker's thought. Freedman considered the communication through depictive gestures a failure in the elaboration of a verbal expression (Freedman, 1972). In the opposite direction, Kendon insisted that gestures are better conveyors of meaning than spoken utterances, which are subject to limitations that do not affect gestures (Kendon, 1986). Indeed, spoken utterances imply a structuration that follows primarily the rules of the language system and implies only indirectly any aspect of the structure of what is being referred to. In contrast, gestures have a direct relationship to the action sequence: motions can describe pictorial diagrams or display spatial relationships or body parts can move to represent actual objects. The gestures can take advantage of many modes of representation, from picturing to lexicalization, which confer more degrees of freedom for expression.

## 2.2 Gesture classification

The previous paragraph not only highlighted how gestural communication is important in human communication but it also described that different speech styles exist. These speech styles present different hand gestures, which change in

motion, meaning and relevance. These different gesture types introduce the issue of gesture classification. The literature presents a plethora of researches that tried to provide a personal solution to the difficult classification problem raised in this field. All the classifications for speech-related hand movements can be included in the pioneering work made by Efron (Efron & van Veen, 1972). Efron's was perhaps the first ethnographic research about gestural communication to combine meticulous participant observation and artistic collaboration with the analysis of everyday social behavior recorded on motion picture film. He conducted not only a study to examine differences in the gestural repertoire of different neighboring immigrant communities (as well as the effect of assimilation on the range of gestures used by their first generation descendants) but he provided also a classification system of the observed gesture types. Figure 2.1 shows some drawings created during this study.

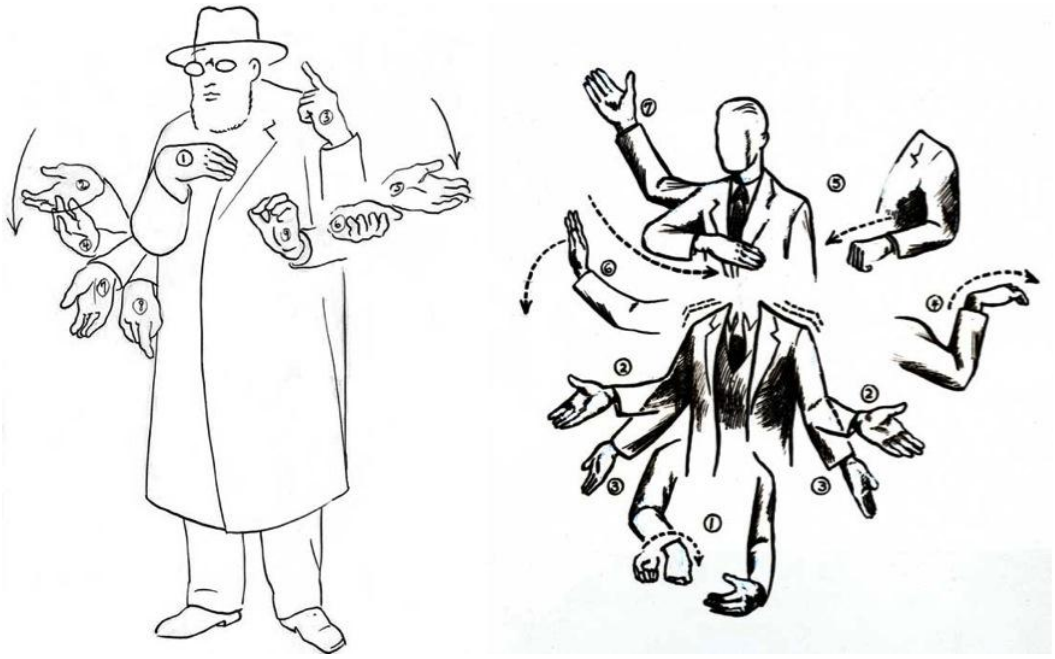

**Figure 2.1 These pictures drawn by Stuyvesant Van Veen illustrate Efron's study about gestures.**

Rimé and Schiaratura took Efron's model as foundation of their model and they review it in order to include all the gesture types emerged also from later researches (Feldman & Rimé, 1991). The whole gesture set can be divided in

three categories: the gestures referring to the ideational process, gestures referring to the object of the speech as depictive type and gestures referring to the object of the speech as evocative type. The gestures referring to the ideational process follow the contour of the speech by marking the speaking person's logical pauses, stresses, and voice intonations. In this category, there are two major subclasses: speech marking hand movements and ideographs. The speech-marking movements comprise the "batonlike gestures", which time the successive stages of the referential activity. This subclass comprises other gestures: the "punctuating movements" that emphasize the speech occurring in bursts and in close coordination with the speech rhythm; the "minor qualifiers" are stylized accentuation movements with a characteristic form but without being staccatolike; the "batonic movements" are used to stress some linguistic item that the speaker wishes to smphasize; the "batons" accent or emphasize a particular word or phrase; the "beats" are simple up-and-down or back-and-forth hand movements with the role of introducing extranarrative elements; "paraverbal" gestures indicate the hand movements that stress the speech intonation or emphasis or that mark the major stages of reasoning. The ideograph class is composed of "logicotopographic gestures" (as defined by Efron) and "metaphoric gestures". The "logicotopographic gestures" are hand or finger movements sketching in space the logical track followed by the speaker's thinking. The "metaphoric gestures" depict some abstract meaning occurring in the speech. In the category of gestures referring to the object of the speech there are two different types: the depictive type and the evocative type. The depictive type gestures are divided in two classes: iconographic or iconic gestures and pantomimic gestures. The iconic gestures are hand movements that parallel the speech by presenting some figural representation of the object evoked simultaneously. In this class, three sub-classes are present. The first is called "pictographic" and describes the shape of the referential object, Rimé and Schiaratura report the example of the upward spiraling movement of the finger to describe a spiral staircase (Feldman & Rimé, 1991). The second sub-class is

called “spatiographic” and is composed of those gestures that represent some spatial relationship regarding the referent; Rimé and Schiaratura described the example of the two open hands placed palm to palm in the person referring to the restaurant located between the bank and the department store (Feldman & Rimé, 1991). The “kinetographic” gestures constitute the third sub-class and describe some action of the referential object; an example is a left-to-right hand movement to represent something moving or passing by. The pantomimic gesture class comprises those hand movements that illustrate the manipulation of objects. In this case, the referred object of the speech is some acting person and the speaker’s hands imitate the described actions. In the evocative type, gestures no longer depict the referent; rather they simply evoke this referent by some action likely to elicit its presence in the common mental space created between the speaker and the listener. There are two sub-classes of evocative type gestures: the deictic gestures and the symbolic gestures. The deictic gestures (called also pointing gestures) consist of hand or finger gestures directed toward some visually or symbolically present object that is simultaneously referred to in the speech. The symbolic gestures (called also emblems) are gestural representations devoid of any morphological relationship with the visual or logical object represented. This type of gestures has very strictly defined characteristics: they have a direct verbal translation consisting of one or two words, they have a precise meaning known by the group, class or culture, and they are the most often used intentionally to send a particular message to the receiver.

The previous paragraph describes the gesture classification presented by Rimé and Schiaratura, and the following schema summarizes the whole system in order to provide a clear overview (Feldman & Rimé, 1991).

Gestures referring to the ideational process

1. Nondepictive gestures: speech markers
    - Stress some elements of the speech for the sake of clarity.

- Parallel the introduction of some new element in the discourse.
- Chunk the sentence following the steps of the underlying reasoning.

Related classes: batonlike (Efron & van Veen, 1972), punctuating movements (Freedman, 1972), minor qualifiers (Freedman, 1972), batonic (McNeill & Levy, 1980; McNeill, 1985), batons (Ekman & Friesen, 1972), beats (McNeill, 1987), paraverbals (Cosnier, 1982).

2. Depictive gestures: ideographs
    - Sketch in space the logical track followed by the speaker's thinking.
    - Parallel abstract thinking.

    Related classes: logicotopographic gestures (Efron & van Veen, 1972), metaphoric gestures (McNeill & Levy, 1980; McNeill, 1985).

Gestures referring to the object: depictive kinds

1. Iconographic or iconic gestures
    - Present some figural representation of the object evoked in speech.
    - Subclass:
        i. pictographic: represents the shape.
        ii. spatiographic: represents some spatial relation.
        iii. kinetographic: represents some action.

    Related classes: physiographic (Efron & van Veen, 1972), motor-primacy representational movements (Freedman, 1972), illustrative gestures (Cosnier, 1982), illustrators (Ekman & Friesen, 1972).

2. Pantomimic gestures
    - "Play" the role of the referent.

Gestures referring to the object: evocative kinds

1. Deictic gestures or pointing
    - Point toward some visually or symbolically present object.
2. Symbolic gestures or emblems
    - Are devoid of any morphological relation with visual or logical referent.
    - Have a direct translation into words.
    - Have a precise meaning known by the group, class, or culture.
    - Usually deliberately used to send a particular message.

## 2.3 Gestures in HCI

In 1980, Bolt introduced his Put-That-There system to the scientific community (Bolt, 1980). He started an important revolution in the Human-Computer Interaction area; in fact, he stated that the machine should understand the human language and not the contrary, making interaction between human and machine more natural. The natural way of interaction between human beings is primarily based on speech and gestures. Therefore, in order to make human computer interaction more natural, a system should get closer to these forms of multimodal communication (Krahnstoever et al., 2002). Bolt developed this system that allowed users to point, gesture, verbally reference "up," "down," "...to the left of...," and so on, freely and naturally, precisely because the users are situated in a real space and can user their natural communication means. The prototype was installed in the Media Room at MIT and Bolt claimed that its physical interface creates a "real-space" environment where the “user's focal situation amidst an ensemble of several screens of various sizes creates a set of geometrical relationships quite apart from any purely logical relationship between any one screen's content and that of any other”. In this proof-of-concept, the virtual graphical space displayed on the screen and the user's immediate physical space in the Media Room converged to become one continuous interactive space that could be naturally manipulated. Bolt’s pioneering work introduced some concepts that still remain very important in

the HCI field. In fact, the Put-That-There system provided a multimodal interface that could recognize human speech and gestures. This means that the interface was based on the two main communication channels that humans use naturally. This last word is also what drove the rest of the development in HCI research.

### 2.3.1 Natural user interfaces

The last decades focused on the human needs in terms of communication and tried to develop interfaces that could be defined natural. Unfortunately, the definition of what is or is not natural remains fuzzy since this issue is not trivial. When people refer to natural user interfaces, they are often talking about interaction modes such as speech or touch. Unfortunately, this is not a comprehensive set of modalities that can define the concept of natural interaction, which should enable users to interact with computers in the way humans interact with the world. The problem is that also the scientific literature did not find a definition or some boundaries that could provide a real classification of natural user interface and have at the same time the consensus of the whole scientific community. The first definition that can be found in a paper dedicated to this issue comes from Valli, who stated that “natural interaction is defined in terms of experience: people naturally communicate through gestures, expressions, movements, and discover the world by looking around and manipulating physical stuff; the key assumption here is that they should be allowed to interact with technology as they are used to interact with the real world in everyday life, as evolution and education taught them to do. " (Valli, 2008). The important element of this definition is the shift of the focus from the modalities to the experience. In fact, the modalities can be more or less opportune in different contexts and usually humans adapt their communication form depending on the surrounding environment. Wigdor and Wixon agreed on this aspect, stating that a natural user interface is “an interface that makes your user act and feel like a natural.” (Wigdor & Wixon, 2011). However, they pushed this concept further including some reflections about usability that are similar to the description present in the ISO/IEC 25010:2011 (coming from the ISO/IEC

9126). In fact, the definition provided by Wigdor and Wixon states that a “natural user interface is one that leads users, at all times, to feel like they are effortlessly interacting with the technology. Novice use is supported, and expert use is the goal. The transition between these is effortless and transparent to the user.” (Wigdor & Wixon, 2011). The first sentence of this definition reflects what already stated by Valli that highlighted the importance of the effortless user experience in the manipulation and interaction. Then, Wigdor and Wixon introduced in this definition the important aspect of learnability. The learnability is a crucial attribute of usability, referring to the ISO/IEC 9126 and ISO/IEC 25010. Usability is the ease of use and learnability of a human-made object, in this case of a system; the ISO defines the usability as "the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use". The aforementioned ISO standards describe the usability as composed of five attributes: understandability, learnability, operability, attractiveness and usability compliance. Specifying that a system should support the user while learning to interact with it in order to make the learning process effortless and easy, which means natural. The first part of this second sentence obviously focuses on the concept of learnability; while the following part highlights that a natural interface should also support the operability. In this case, the operability is intended as the propriety of a system to make easy and more efficient the use of the system by an expert user; this concept expresses what happen with the manipulation of real objects, the user becomes more and more skilled in the use of determined artifacts or in the performance of repeated movements with the continuous training. The last part of this definition focuses on the seamless and effortless passage from the novice user status to the expert user status in a transparent way, this mainly thanks to the understandability. The attractiveness is the reason that should make the user approach the interface. These very same concepts can be also found in Nielsen’s definition of usability, which defines the usability as

composed of five quality attributes: learnability, efficiency, memorability, errors and satisfaction (Nielsen, 2003).

Natural user interfaces are those that enable users to interact with computers in the way humans interact with the world (Jain et al., 2011). The interaction modes used by humans comprise the speech, gestures, eye gaze, facial expressions and a full spectrum of interactions involving the many human senses. However, until now the research and design of natural user interfaces focused on two modalities: speech and gesture (Wachs et al., 2011). Although vocal communication is the preferred modality by individuals having developed linguistic skills, speech has been revealed as inconvenient for many applications in HCI. Shneiderman stated that human-human relationships are rarely a good model for designing effective user interfaces (Shneiderman, 2000). In fact, spoken language is effective for human-human interaction but often has severe limitations when applied to HCI. One of the most important issue is due to the fact that speech is slow for presenting information, is transient and therefore difficult to review or edit (Shneiderman, 2000). Moreover, the vocal communication implies other physical problems such as fatigue from speaking continuously and the disruption in an office filled with people speaking (Shneiderman, 2000). In the last case, the inconvenience can come from the dazing commotion of the people in the same office all together just to interact with their PCs, which creates also a noise that could degrade the system performances (for this reason many researches treat exactly this point (Schuller et al., 2009; Gong, 1995)). Shneiderman highlighted another important issue of vocal interaction, which derives from the interference that speaking creates with other cognitive tasks (Shneiderman, 2000). In fact, the activities of speaking and saolving complex problems solicits the same areas of the human brain. In fact, short-term and working memory are sometimes called acoustic or verbal memory. The part of the human brain that transiently holds chunks of information and solves problems also supports speaking and listening. Therefore, working on tough problems is best done in quiet environments—without

speaking or listening to someone. However, because physical activity is handled in another part of the brain, problem solving is compatible with routine physical activities like walking and driving. In short, humans speak and walk easily but find it more difficult to speak and think at the same time. Similarly when operating a computer, most humans type (or move a mouse) and think but find it more difficult to speak and think at the same time. Hand-eye coordination is accomplished in different brain structures, so typing or mouse movement can be performed in parallel with problem solving. However, speech has proved useful in some particular contexts: for store-and-forward messages, alerts in busy environments, and input-output for blind or motor-impaired users, or in the automotive field. In particular, voice enabled dialog systems are well suited for in-car applications, where the user must maintain the focus on the road while interacting with the car. In fact, driving is an eyes-busy and hands-busy task and the only wideband communication channel left is speech. Such systems are in the midst of a transformation from a cool gadget to an integral part of the modern vehicles. This modality became so important in the automotive that currently it is possible to find commercial solutions; for example, one of the first speech enabled infotainment systems in mass production cars was the Microsoft Blue&Me deployed initially on selected FIAT models (Tashev et al., 2009).

### 2.3.2 Gestural interfaces

Differently from the spoken language, gestural communication is used not only for the human-human interaction but also for the interaction with the surrounding environment (Wachs et al., 2011). Moreover, gestures help the memorization of information thanks to the sense of proprioception and the motor processes related to the gestures, as explained by Rimé and Schiaratura (Feldman & Rimé, 1991). The gestural interfaces have many useful applications; Mitra and Acharya mention the following examples: developing aids for the hearing impaired; enabling very young children to interact with computers; designing techniques for forensic identification; recognizing sign language; medically monitoring patients' emotional states or stress levels; lie detection;

navigating and/or manipulating in virtual environments; communicating in video conferencing; distance learning/tele-teaching assistance; monitoring automobile drivers' alertness/drowsiness levels (Mitra & Acharya, 2007). Wachs et al. mention other examples such as accessing information while maintaning total sterility (or avoiding touching devices with dirty hands), overcoming physical handicaps, exploring big data and communicating with robots (Wachs et al., 2011). Another important domain of application for gestural interfaces is domotics (de Carvalho et al., 2013). However, it is possible to notice that the aforementioned examples of application involve many different kinds of gestures. Therefore, it is necessary to find a definiton of gesture. Kendon accepts the definition proposed by Studdert-Kennedy, who wrote that a gesture is "an equivalence class of coordinated movements that achieve some end" (Studdert-Kennedy, 1987). Since this definition comes from the psychology domain, it remains quite generic; even if, Kendon applied this very same definition to the reduced area of hand gestures (Kendon, 2000). This definition changes when applied to the HCI field, where it acquires a definition more technology-oriented. In fact, Karam stated that "the term gesture is used to refer to an expansive range of interactions enabled through a variety of input technologies, devices, and strategies (including computer-vision, data gloves, or touch screens), to control tasks in application domains such as virtual reality, robotics, pervasive, and ubiquitous computing. However, there exists no single theoretical perspective that can support a common discourse when considering gestures as a field of interaction techniques" (Karam, 2006). LaViola provided two definitions in order to cover the duality of the gesture description. The first definition is near to the psychological concept: "gestures are movements with an intended emphasis and they are often characterized as rather short bursts of activity with an underlying meaning" (LaViola, 2013); the other one is more technical for the explicit application in computer science: "a gesture is a pattern that can be extracted from an input data stream" (LaViola, 2013). Although from these definitions it is clear that a gesture is a movement of a part of the body, the

underlying meaning of a gesture can be expressing an idea or manipulating a physical object or both. Moreover, the gestures that are used in the human-human communication can have different styles and can be classified in many different classes; the same happens in the HCI field. Since HCI research used to lack a commonly used system for describing gestural interaction, many HCI researchers needed to use a classification system for their works and the many of them (Wexelblat, 1998; Quek et al., 2002; Eisenstein & Davis, 2004; Kettebekov, 2004;) referred to Kendon's model that is reported in the previous paragraph in the version revised by Rimé and Schiaratura. In order to fill this gap, Karam presented in her PhD thesis a comprehensive work based on the literature review for the gesture classification. Karam obtained this classification extending the model previously provided by Quek et al. (Quek et al., 2002). Quek's model can be summarized with the following schema (Quek, 2004):

- Manipulative
    - Deictic
    - Others
- Semaphoric
    - Static
    - Dynamic
- Conversational
    - Emblems
    - Iconic
    - Deictic

Karam's extended model can be schematized as follows:

- Deictic Gestures
- Manipulative Gestures
    - 2D
    - 3D
    - Tangible gestures and digital objects

    - Tangible gestures and physical objects
- Semaphoric Gestures
    - Static
    - Dynamic
- Gesticulation
- Language Gestures

Karam created this model (Karam, 2006) taking as base Quek's model (Quek, 2004) and extending it according to the literature review concerning gestural interfaces. In this model deictic gestures are the first mentioned class. Deictic (or pointing) gestures represent a class that sits between the manipulative gestures and the gesticulation. Quek et al stated that as in the case of manipulative gestures, deictics have the capacity of immediate spatial reference (Quek et al., 2002). In particular, Karam claimed that free hand pointing gestures are a very natural way of interacting with objects and tools (Karam, 2006). The study of multimodal natural interaction paradigms combining speech and gestures has a long history starting in 1980 with the above mentioned Bolt's Put-That-There system (Bolt, 1980). Deictic gestures are used jointly with isolated word commands allowing the user to point at a location on a large screen display in order to locate and move visual targets. Among the different communication modalities, the recognition and interpretation of deictic gestures is of great importance not only in the ambient intelligence field but also for more general HCI applications. For instance, human-robot interaction is a very important application domain. Droeschel et al. proposed a time-of-flight camera-based approach for person awareness and for the detection of pointing gestures for a domestic service robot (Droeschel et al., 2011). The robot estimates the pointing directions and matches the shown objects in order to recognize the target of the pointing gesture and the user's intention. Moreover, in some contexts it is easier and more accurate to point an object than to give a verbal description of the object and its location. Starting from this motivation Kahn et al. demonstrated the use of pointing gestures to locate objects in a test environment. Their system

operates on various feature maps (intensity, edge, motion, disparity, color) (Kahn et al., 1996). Whereas Jojic et al. detected and estimated pointing gestures solely in dense disparity maps and depth information, since color-based approaches are sensitive to lighting changes and the clothing worn by the user (Jojic et al., 2000). Also Nickel et al. used color and disparity information to recognize and understand the pointing gesture (Nickel & Stiefelhagen, 2007). Cipolla et al. used the position and the shape of the hand, (e.g. the index finger) from different views in order to locate the pointing destination on a 2-dimensional workspace (detected through uncalibrated stereo vision with active contour) (Cipolla & Hollinghurst, 1996). Unlike these approaches, in which sensors are external to the user (they are placed in the environment or on a robot), other solutions investigated the possibility of mounting the technology on the user. Carrino et al. proposed a solution based on sensors that can be worn or hand-held (Carrino et al., 2011). In this work, the pointing direction and the user's position in the environment was calculated through the Parallel Tracking and Multiple Mapping (PTAMM) method. This method was introduced in the seminal researches and results presented by Klein and Murray; in particular, the results about localization, tracking and mapping can be found in (Klein & Murray, 2007; Castle et al., 2008). In another work (Castle & Murray, 2009), also an interesting object recognition approach for augmented reality applications using PTAMM is presented.

With reference to Quek's defintion, manipulative gestures are defined as "those whose intended purpose is to control some entity by applying a tight relationship between the actual movements of the gesturing hand/arm with the entity being manipulated" (Quek et al., 2002). A manipulative gestures involve complicated interactions requiring interpretation by the computer system. Karam categorized the manipulations in four different types. The first type encompasses the 2D manipulations, which usually occur within 2D displays for controlling graphic objects or windows. The traditional concept of 2D manipulation refers to direct manipulations performed with the mouse; however, Rubine stated that a

manipulative gesture differs from the mouse manipulation because of the provision of parameters to the system, indicating the nature of a transformation or relocation of the digitally rendered object (Rubine, 1992). Wu and Balakrishnan demonstrated 2D manipulative gestures for table top surfaces fitted with electronic material (Wu & Balakrishnan, 2003). Similar work by Rekimoto used manipulative gestures drawn from actual table top gestures such as sweeping and isolating groups of objects with the hands to manipulate them (Rekimoto, 2002). With reference to technology, the most popular approaches are vision-based; it is possible to use cameras on the top of the table (Malik & Laszlo, 2004) or behind a semi-transparent surface (Wilson, 2004). Similar systems can also employ 3D gestures adding the third dimension through different technological approaches as pressure sensors or 3D cameras. The 3D gestures can enable more complicated manipulations. For instance, one of the first systems adopting this concept can be found in Minsky's work, where this finger painting application senses the pressure exerted on the display to indicate line thickness (Minsky, 1984). 3D manipulative gestures are also used to identify and transfer digital objects between different devices, a first example can be found in the Pick and Drop gestures presented by Rekimoto in (Rekimoto, 1997). A more recent work, called LightSpace, has been developed at Microsoft Research and presented a similar concept but enhancing the gestures possibilities thanks to the adoption of advanced technology. In this work, Wilson and Benko presented a similar concept for the transfer of digital content among different interactive surfaces. These surfaces allow performing multi-touch interactions on a virtual object, the user may transfer the object to another display by simultaneously touching the object and the destination display. Or the user may "pick up" the object by sweeping it into their hand, see it sitting in their hand as they walk over to an interactive wall display, and "drop" the object onto the wall by touching it with their other hand. Besides the manipulations of 2D objects through the touch gestures, there are the tangible gestures for the manipulation of tangible objects for different purposes. Van den Hoven and

Mazalek defined the tangible gesture interaction as “the use of physical devices for facilitating, supporting, enhancing, or tracking gestures people make for digital interaction purposes. In addition, these devices meet the tangible interaction criteria”. In (Van den Hoven et al., 2013), it was stated that tangible gesture interfaces must meet tangible interaction criteria by: 1) interacting with the physical objects, 2) using physical and cognitive skills, 3) for digital interaction purposes. These three criteria constitute the foundations of tangible interface design. Tangible objects that are used as computer input are often referred to as gestures. In (Hinckley et al., 1998), it was presented an interaction where the physical manipulation of a doll’s head is displayed onto a screen as the tomography of a human brain. Another transdisciplinar example is represented by the ArcheoTUI, which is a new tangible user interface for the efficient assembly of the 3D scanned fragments of fractured archeological objects (Reuter et al., 2010). This system allows the user to use tangible props for the manipulation of the virtual fragments. A famous system that merged the object manipulation with the augmented reality and made the computational and physical manipulations converge is represented by Urp (Underkoffler & Ishii, 1999). The Urp allows physical architectural models placed on an ordinary table surface to cast shadows accurate for arbitrary times of day; to throw reflections off glass facade surfaces; to affect a real-time and visually coincident simulation of pedestrian-level windflow; and so on. Another interesting example of tangible gestures can be found in (Schloelmer et al., 2008), the authors used a Nintendo Wii controller as device for the interaction with a screen (tangible gestures for virtual object manipulation). Using the same technology and similar tangible gestures, a system was implemented for the manipulation of tangible objects, i.e., the control of a robot in the environment (Guo & Sharlin, 2008). The third class encompasses the semaphoric gestures, which have been comprehensively defined by Quek et al.: “semaphores are systems of signaling using flags, lights or arms. By extension, we define semaphoric gestures to be any gesturing system that employs a stylized dictionary of static or dynamic hand or arm gestures.

Semaphoric approaches may be referred to as communicative in that gestures serve as a universe of symbols to be communicated to the machine". In (Karam, 2006) Karam stated that according to (Wexelblat, 1998) and (Quek et al., 2002), semaphoric gestures are frequently discussed in the literature as being one of the most widely applied, yet least used, forms of human gesturing. In particular, semaphoric gestures are still seen as a practical method of providing distance interactions for smart rooms and intelligent environments. These kinds of gesture can be separated in two different sub-classes: the static gestures and the dynamic gestures. The static gestures are static poses, typically of the hand, that can be associated to the symbolic gestures (or emblems) in Rimé and Schiaratura's model (Feldman & Rimé, 1991). For example, joining the thumb and forefinger to form the A-OK sign is a static pose and should be classified as an emblem, in this case, defined as a static semaphoric gesture. These gestures can be used to command a robot using hand postures (as proposed by (Yin & Zhu, 2006) or to control home appliances through static finer gestures (as proposed in (Jin et al., 2012). The dynamic gestures are dynamic movements of the hand or fingers or head, and these body part movements draw a path in the air (see the pathic information described in (Mitra & Acharya, 2007). In (Henze et al., 2010), a curious study was performed where they analyzed static and dynamic gestures as forms of interaction with a music playback application. These authors performed a 3-step evaluation with twelve users with different profiles, five male and seven female. The results indicated that dynamic gestures are easier to remember, more intuitive and simpler for controlling a music application. Of course, these results should be limited to this application and could be biased by socio-cultural factors. According to Karam's model, stroke gestures such as those executed using a pen or a finger are also considered semaphores (Karam, 2006). These gestures are usually executed on a surface with a pen (e.g., Delaye et al., 2011; Lenman et al., 2002; Zao & Balakrishnan, 2004; Hofer & Kunz, 2010) or a finger (e.g., Jiang et al., 2012; Bragdon et al., 2011) or using a hybrid approach that allows using both (e.g., Hinckley et al.,

2010; Frisch et al., 2010). An example that is worth to mention can be found in (Ishii's et al., 2009), where the same concept of performing stroke gesture has been applied to the 3D space of a real environment through a laser-pointer. In this case, the pathic information drawn by the user was used to infer specific commands to a robot. The fourth category of gestures in Karam's model is dedicated to gesticulation that Wexelblat defined as idiosyncratic, not taught, empty handed gestures that are considered for directive style interfaces (Wexelblat, 1995). Karam included many different types of gestures coming from Rimé and Schiaratura's model in the gesticulation category. In fact, she stated that all the gestures that add clarity to speech recognition as ideographs and depictive ones (both iconographic and pantomimic) are part of gesticulation. These gestures can be used in human-computer interaction for many different purposes: for example, for generating multimodal feedback information about route directions (Striegnitz et al., 2005), for creating more usable and pleasurable interfaces (Lee et al., 2013), and for emotion recognition (Gunes & Piccardi, 2007). The last category includes all the sign languages. A sign language is defined by the Marriam-Webster as "a formal language employing a system of hand gestures for communication (as by the deaf)". Actually, many types of sign languages exist, e.g. the American Sign Language, the British Sign Language et cetera. In fact, the last the 2013 edition of Ethnologue lists 137 different sign languages (Lewis et al., 2013). In the human-computer interaction field, there has been an interest in developing sign language interpreter based on artificial intelligence (e.g., Chai et al., 2013; Starner et al., 1998) but Cooper et al. pointed out that it knew a slower evolution if compared to the speech recognition (Cooper et al., 2011).

### 2.3.3 Gesture design

Although Karam's model is based on the literature review (Karam, 2006), the gesture taxonomies implemented in many systems cannot be categorized in only one category. Many systems provide a vocabulary of mixed gestures. It is not clear if this happens because of the limits imposed form the technology or

because of the intrinsic nature of human communication, which is composed of many different gesture types that are used in different contexts and without a strict paradigm of application. Hence, the literature presents a plethora of heterogeneous systems that adopt specific taxonomies creating confusion. As Norman pointed out, the "gestural systems are one of the important future paths for a more holistic, human interaction of people with technology" but they need to be standardized (Norman, 2010). Since the standards that have been created for the GUIs are not directly applicable for the NUIs, gestural systems require novel methods for design and development. In the current situation, the gestural interfaces are in their infancy and for this reason they are undergoing a phase of exploration where the gesture design is left to the developers' inspiration and creativity. This wild development led to some "usability disasters", as Norman and Nielsen defined it in (Norman &Nielsen, 2010). They mentioned also some examples to explain what they meant with this expression and to describe some of the different types of usability mistakes that can be made. For example, a gesture that is been adopted in many systems is the pinching to change image scale. Unfortunately, this is not a standard and it could happen that changing operating-system or application can provide a different way to scale the image. In some applications, the scale can be changed through plus and minus boxes, others allow flipping the screen up and down or maybe left and right. Sometimes, the images can be touched and even in this case the triggered action can vary depending on the application: opening a hyper-link, unlocking the image to move it or just enlarging the image. Many operating systems provide some guidelines for the gestural application development but unfortunately they differ from one another creating this lack of consistency. Other gestures are getting such a popularity that are gradually acquiring a universal meaning. For instance, the shaking an MP3 player or a phone for the "randomize" action: when a user does not like the current status, e.g., the current track, and wants to make the device propose a new random option, he has only to shake the device. This gesture, which has been discovered

accidentally, feels natural and is fun. However, shaking a smartphone or an MP3 player can be handy but shaking a large tablet is neither easy nor much fun. Norman stated also that although gestural interfaces adopting wide gestures provide benefits as the enhancement of the pleasure in using it and in fostering the physical activity, they present serious side effects. These interfaces, if not appropriately designed, can do damages. Norman's example depicted the introduction of the Nintendo Wii (Norman, 2010). One of the basic games included with this innovative gaming console based on gestural interaction was the bowling game. In this game, the user had to swing the arm as if holding a bowling ball, and then, when the player's arm reached the point where the ball was to be released, to release the pressure on the hand-held controller's switch. This metaphor for the natural interaction was based on the movement that accompanied the swing and the release of the ball like in the actual sport. Unfortunately, in the heat of the game many players would also release the hand from the controller losing the grip of the remote controller and throwing it forward, often towards the television breaking its screen. Since this problem occurred so frequently, Nintendo added a wrist strap to avoid the controller being thrown. This expedient resolved the broken TV issue but did not fix the usability problem that was at the foundation of the interface. Therefore, Norman and Nielsen provided seven fundamental principles of interaction design that should be respected a priori, that means independently from the technology. As reported in (Norman & Nielsen, 2010), the seven principles are: visibility (also called perceived affordances or signifiers), feedback, consistency (also known as standards), non-destructive operations (hence the importance of undo), discoverability, scalability and reliability. These principles are directed to the gestural interface designers but they do not provide a real method for the design process. In (Wachs et al., 2011), a list of basic requirements for the development of vision-based hand-gesture applications is reported. They mentioned the price, the responsiveness, the user adaptability and feedback, the learnability, the accuracy, the low mental load, the intuitiveness, the comfort, the lexicon size,

the “come as you are”, the reconfigurability and the interaction space, the gesture spotting and the immersion syndrome, and the ubiquity and wearability. Some of these requirements concern the technological development of the systems but other are directed to the gesture design. Especially, the learnability, the low mental load, the intuitiveness, the comfort and the lexicon size are characteristics that the gesture designers have to consider in order to develop a good interface. The learnability has been already defined and its usefulness highlighted. The low mental load concerns the user’s effort in recalling the gesture trajectories, finger postures and associated actions; another source of mental load is due to the vision occlusion by the inconvenient hand position. The intuitiveness is correlated to the learnability and refers to the cognitive association of the gesture with the performed action; this principle involves the use of the gesture classification coming from the psychology in order to select the natural gestures already used by people to communicate. The comfort concerns biomechanical ergonomics and suggests that the designers have to take into account that a gesture should not require intense muscle tension over long periods, a syndrome commonly called “Gorilla arm”. Wachs et al. mentioned the lexicon size but only because the classifier performance degrades as the gesture number increases; actually they stated that hardly any literature exists on user performance as a function of gesture vocabulary size. In contrast, some researchers claimed that the size does affect the cognitive load (e.g., Nielsen et al., 2004; Kela et al., 2006); also Cabral et al. stated that reducing the number of gestures increases the usability and reduces the duration of the training phase (Cabral et al., 2005). For all these reasons, the taxonomy size has been included in the principles for the gesture designers. The aforementioned principles are general guidelines but they do not constitute a method that could help designers to find the opportune gesture in order to increase these qualities of their interfaces. Hence, some researchers suggested a smart approach to gesture design based on observation and the use of methods as the “Wizard of the Oz” or role playing (e.g., Nielsen et al., 2004; Akers, 2007). The Wizard-of-Oz

approach consists of eliciting gestures from non-technical users by first portraying the effect of a gesture as demonstrated by an unseen operator manipulating the system (called wizard) and then asking the users to perform its cause, i.e., the associated gesture that should trigger that command (Wobbrock et al., 2009). The role playing involves preparing scenarios that implement the types of messages related to the target functions previously selected. Then the researchers will reinterpret real-life situations and observe the gestures performed by the non-technical testee; the observation usually involves also recording the testee's gestures with some cameras for later analysis. Another philosophy recently emerging suggests letting the user to design and choose personalized gestures. In fact, a recent study found out that user-designed gestures improve the memorability, which is a very important characteristic of gesture sets. In (Nacenta et al., 2013), the authors empirically tested and analyzed the difference in terms of memorability between pre-designed gestures and user-defined gestures. Their findings make them claim that user-defined gestures are easier to remember, both after creation and on the next day; moreover, the test participants preferred performing user-defined gestures and they think that they take less time to be learnt. This approach introduces a new challenge: being able to provide an interface that can guide the user in designing their self-defined gestures (Oh & Findlater, 2013).

The aforementioned approaches are valuable methods for the gesture design that should be applied in order to achieve the qualities of gestural interfaces presented in (Wachs et al., 2011) and (Norman & Nielsen, 2010). These approaches aim at facilitating the design of a gesture that can provide the affordance to improve the usability of the whole system. In particular, some methods focus on improving the memorability of gestures, e.g., self-defined gesture method proposed in (Nacenta et al., 2013), others to improve the learnability referring to cultural cues, e.g., role playing method presented in (Nielsen et al., 2004). These methods take also into account the function that should be associated to the particular gesture but the risk introduced by

following directly these processes is that there will be one gesture for each function. Since the number of functions is often high, this means that the gesture taxonomy associated to the system will be very vast and, therefore, hard to learn for the user. In this thesis, a novel method for the gesture to function mapping is presented introducing the concept of functional gestures. This method allows reducing the number of different gestures to be associated to the functions in order to create gesture taxonomy that can be easily learnt. This method implies that an opportune organization of the context information in order to optimize the number of the gesture taxonomy and the associated functions. Since usually the contextual information is not provided by a single source, it is necessary to introduce an overall framework that allows opportunely organizing the information on multiple layers: from the lowest level where the information provided by the sensors to the highest level where the system could suggest the action to the user. This framework should not be technical on the programming level but rather a theoretical structure that could be applied in every system in order to facilitate the gesture recognition process independently of the adopted technologies.

## 2.4 Gestural interfaces in the ubiquitous computing era

An approach widely used for classification of gestural interfaces is based on the technological paradigm adopteed for the system development. A popular example can be found in Karam's dinstinction into two categories: perceptual and non-perceptual interfaces (Karam, 2006). A perceptual input is defined as the recognition of gestures without the need for any physical contact with an input device or any physical objects. This classification system has been used also by other researchers (de Carvalho et al., 2013) but this categorization paradigm cannot be fit for the gestural interfaces of the ubiquitous computing era. In this era, the users move in smart environments, which have been defined in (Cook & Das, 2004) as a technological concept based on Weiser's vision of "a physical world that is richly and invisibly interwoven with sensors, actuators, displays, and

computational elements, embedded seamlessly in the everyday objects of our lives, and connected through a continuous network" (Weiser et al., 1999). In this context, sensors and actuators are distributed everywhere: they are in the walls, furniture, artifacts, clothes et cetera. They can be able to provide both interaction modes: contact and distance gestures. This factor indtroduces a problem in the calssification based on the perceptual and non-perceptual categories. In addition, the question of smart clothes classification throws a new thorny problem in. In fact, the clothes are worn by the user and they can sense the gestures that do not involve any obejct manipulation but the textile is in contact with the user's body. For these reasons, in the ubiquitous computing vision it could be more appropriate to use a different classification system based on other technological categories. Moreover, the need of this different classification system acquires more importance when considering the deep influence that the Weiser's vision had on the whole research conducted in the last two decades. It is possible to verify this performing a research in the most important repositories of scientific papers for the computer science domain. Figure 2.2 depicts the number of papers mentioning "ubiquitous computing" in the digital libraries of the Association for Computer Machinery (ACM), the Institute of Electrical and Electronics Engineers (IEEE) and Elsevier (using the DirectScience repository); in addition, the graph shows also the number of papers found in Google Scholar, which provides a larger overview since it gathers the publications from many different international repositories. The number of papers are reported for each year since 1991, the year of the publication of Weiser's seminal article intitled "The computer for the 21st century" in the "Scientific american" journal (Weiser, 1991). This article was the first official presentation of the concept of ubiquitous computing developed at the Xerox PARC to the research community (Weiser et al., 1999). The numbers reported in the graph clearly shows that this vision not only became very popular and its escalation during the noughties but that its popularity is still growing.

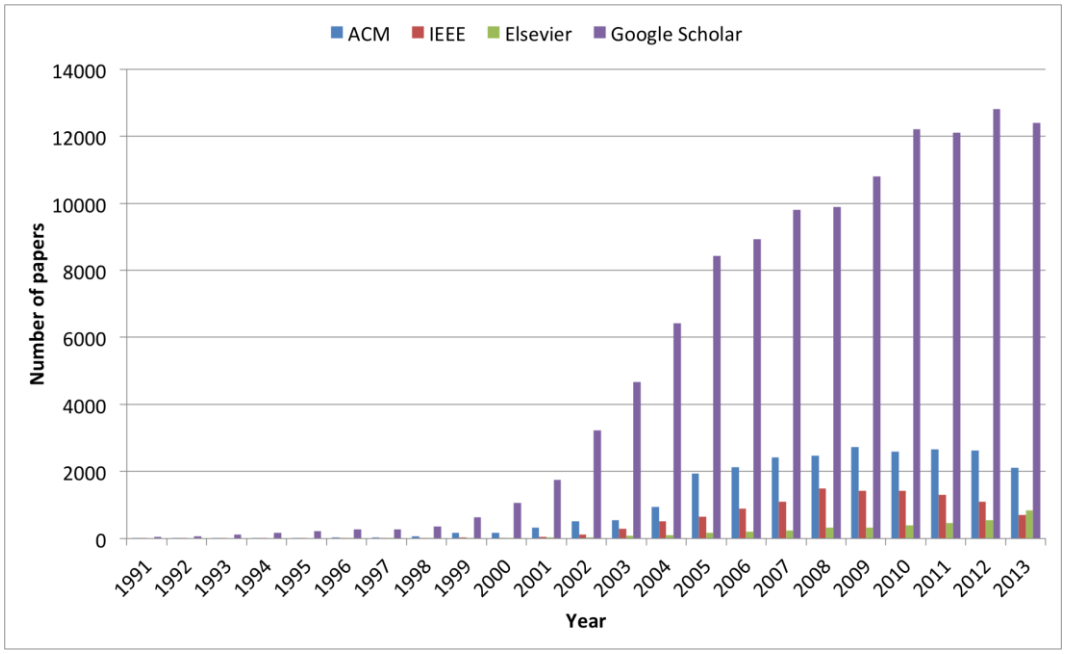


**Figure 2.2 The influence of ubiquitous computing vision on the computer science research expressed as number of papers mentioning it.**

During the last decade, ubiquitous computing ceased of being only a vision and became a new field of computer science. For instance, the 1999 was the year of the first edition of the conference on ubiquitous computing that currently is one of the largest ACM conferences and is called ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp) with thousands of researchers attending it every year. Many academic journals devoted to this field, a couple of particularly important journals are "Personal and Ubiquitous Computing" (edited by Springer), "Pervasive Computing" (edited by IEEE), and "Pervasive and Mobile Computing" (edited by Elsevier). The discipline of ubiquitous computing has spread so widely that Abowd wrote a paper arguing that "ubiquitous computing, the third generation of computing, is here and no longer requires special attention, as its ideas and challenges spread throughout most of computing thought today" (Abowd, 2012). Abowd stated that ubiquitous computing as intellectual area broadened to the point of disappearance. In fact, it permeated the computing universe so deeply that it makes no longer sense speaking of ubiquitous computing as a research topic; Abowd suggested that ubiquitous computing actually represents the intellectual domain of all of

computing and that it made the current computing climate different as it has ever been.

### 2.4.1 The ubiquitous computing paradigms

In the ubiquitous computing era, sensors are distributed everywhere. From an interaction design point of view, in the ambient intelligence scenario two main actors are present: the user and the environment (Augusto, 2010). Therefore, two obvious paradigms exist for the development of human-environment interaction systems: integrating the technology in the environment and on the user (Pentland, 1998). The environmental computing paradigm uses distributed sensors and actuators in the interaction space to detect and react to the inhabitant's movements, gestures, and activities (Cook & Das, 2004). Wearable computing technologies place them onto the body as wearable garments or portable accessories (Mann et al., 2002). Ubiquitous computing systems are where smart environments and wearable computing meet (Kocaballi, 2010); cobining the environmental and wearable paradigms generates a third hybrid paradigm that can be defined as pervasive computing (Figure 2.3).

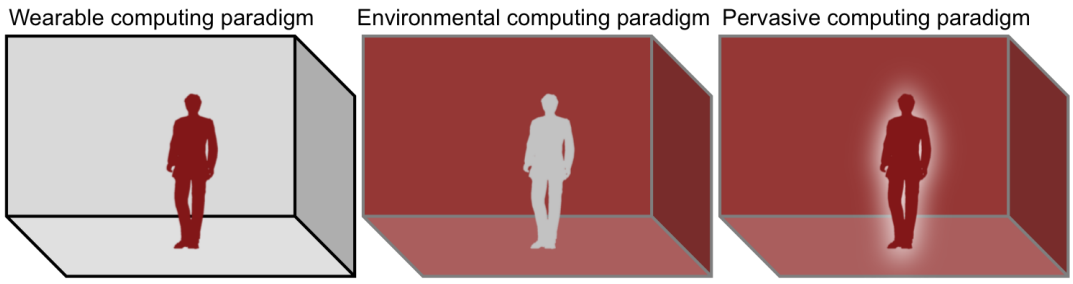


**Figure 2.3 The three computing paradigms: the red elements shows where the technology is embedded. On the left, the wearable computing paradigm implies the distribution on the user ; in the center, the environmental paradigm implies the technology distributed in the environment ; onn the right, the pervasive computing paradigm is obtained when combining the other two paradigms.**

With particular reference to the development of gestural interfaces for the human-environment communication, environmental and wearable paradigms have different strong points and weaknesses.

### 2.4.2 The environmental computing paradigm

The environmental computing systems are non-cumbersome and allow free-hand interaction (Baudel & Beaudouin-Lafon, 1993). Many gesture recognition systems adopting the environmental paradigm use cameras as sensors. For example, Sato et al. used multiple cameras distributed in the environment to detect deictic gestures for the communication with an intelligent home service robot (Sato et al., 2007). A similar project for deictic gesture recognition in a smart home has been developed in (Do et al., 2006). Information extraction from 2D video streams involves many limitations because gesturing in a real room needs three dimensions for a complete representation, for this reason often it is necessary to use multiple calibrated cameras. However, some techniques for gesture recognition using uncalibrated cameras exist; unfortuntately, with this kind of solutions the recognition accuracy severly degrades with the position and the direction.  and often In fact, the authors of [9] utilized stereo-cameras to extract depth information from disparity map. Using 2D cameras entails other important problems as background substraction and reselience to lighting changing. A valid alternative can be found in the adoption of 3D cameras as time-of-flight cameras (Droeschel et al., 2011) or with structured light (Kim et al., 2011), which avoid these problems. Other solutions exploit the potentiality coming from the fusion of different types of sensors as microphones and 3D cameras (Fleer & Leichsenring, 2012). The environmental computing paradigm is often limited for the interaction area. In fact, the dimensions of the interactive area can depend on many factors as the number of sensors or the peculiarity of the environment; for example, it is easier to track users in indoor environments, where the interaciton area is limited and usually there is less noise.

### 2.4.3 The wearable computing paradigm

Pentland and his team at MIT chose to change paradigm back in the early nineties. In fact, they claimed that moving from the environmental paradigm to the wearable one meant changing from the third person perspective to first-person; that allowed developing systems that could be more intimately and

actively involved in the user's activities (Pentland, 1998). The wearable systems seemed being perfect for the implementation of gestural interfaces since the integration of sensors in the clothes can enhance the body movements detection. One of the first wearable systems was developed by Starner et al. and used a head-mounted camera for the ASL gestures (Starner et al., 1998). Another interesting example of wearable system is Starner et al.'s pendant: an infrared camera with LED emitters embedded in a jewel wearable as a necklace or a pin that can recognize gestures to control home appliances (Starner et al., 2000). The aforementioned examples of wearable computing systems were focused on the recognition part, other works demonstrated that this sort of interfaces can integrate also displays. For instance, the Wear Ur World project (Mistry et al., 2009), better known as SixthSense (Mistry & Maes, 2009), embeded a camera for gesture recognition and a small projector to display personal information on different surfaces. A similar concept has been developed in the OmniTouch project where a 3D camera was used for the gesture recognition (avoiding the use of markers as it was necessary for the SixthSense project) and adopting a pico-projector for the information visualization on many different kinds of surfaces (Harrison et al., 2011). The same display technology was implemented in the Skinput system, where it was possible to interact only on the user's body thanks to a peculiar acoustic technology (Harrison, 2010). Also in this paradigm it is possible to combine different sensors for the gesture recognition as in Carrino et al.'s system where a camera using a SLAM algorithm for the direction recognition and intertial sensors for the gesture recognition was used as an interface for the interaciton with the smart environment (Carrino et al., 2011). Recently, also the industry paied a special attention to the development of wearable systems. A resounding example is the project Glass concucted by Google, which is a pair of glasses with a head-up display and an embeded camera; it is possible to interact with it through gestures simply touching the side of the side of the camera with a finger or moving up the head or via speech recognition (Starner, 2013). Also the smart watches and bracelets became very

popular both in the academia and the industry; many systems have been released both for activity monitoring or gestural interaction but they usually need to be connected to a smartphone to overcome computational limitations (Angelini et al., 2013). In fact, the wearable computing paradigm has to deal with the classic issues of mobile devices, namely encumbrance, performances, energy consumption and wireless communication limits.

### 2.4.4 The pervasive computing paradigm

The pervasive computing paradigm is the combination of the environmental and wearable computing paradigms, which have typically been combined to overcome technical limitations like reducing the computational complexity, increasing the effectiveness, or resolving privacy issues (Rhodes et al., 1999). In fact, the environmental paradigm tends to have difficulties with privacy and personalization. For personalization, every time a person interacts with a new environment, it is necessary to transfer the personal data to the environmental system. This process is quite annoying and leads to the other problem. Sharing personal data with environmental infrastructure cannot guarantee a correct use of them. On the other hand, the wearable computing paradigm is perfect for personalization. Since the wearable system is always on the user the personla profile never needs to be transferred to a new environment. However, the wearable computing paradigm presents troubles with localized information, localized resource control and resource management between multiple people. The localized information implies that when some information is updated in a local environment, then every wearable system needs to be given the new information and it is not possible to access data networks everywhere, yet. The localized control implies that the wearable system should be able to control resource off the person's body. This means using the hardware of the wearable device, which has limited capabilities in terms of computational power and energy consumption. That counts also for the resource management when multiple users are involved, in fact sharing information implies an intense hardware load and accessing to the control of the same local resource can create

conflicts or an inadequate control. A pervasive computing paradigm that properly combines the environmental and wearable computing allows alleviating these issues (Rhodes et al., 1999; Carrino et al., 2011). In fact, the wearable system can contain the personal information allowing the user to access to it without having to enter it in every environmental system. At the same time, the wearable and environmental systems can share the computational load in order to optimize the resource management for an enhanced experience for the control of the localized resecources and without the necessity to update the wearable system in case of changes of the environmental system. Some examples of system using sensors distributed on the user and connected to the smart environment can be found in the literature (Neßelrath et al., 2011; Kivimaki et al., 2013). In this case, the computational part is completely delegated to the smart environment and the personalization is missing. In other systems, the gesture recognition was committed to the wearable device and the activation to the environment (Kühnel et al., 2011; Wu et al., 2010). In these examples, the personalization is maintened and also the local control and information but the computational part is competly demanded on the wearable device, which is not optimal. On all the previously mentioned systems that adopts the pervasive computing paradigm, the smart environment was deprived of sensors that were on the user. In the literature, it is possible to find some examples where the sensors are distributed in both the wearable and environmental systems. In (Budde et al., 2013), the user had to interact with the home appliances pointing at them and give a command. The pointing was detected through a 3D camera and the gesture recognition was managed by the environmental system; at the same time, the command was given through the smartphone. In this example, both the environmental and wearable systems were perceptual and that allowed to alleviate all the issues that usually should affect the two systems separately. However, in that system the interaction was split in two phases, where the first part was executed by the smart environment and the second one by the smartphone. It is possible to distribute sensors both

on the user and in the environment to recognize the same command at the same time. For instance, in (Wenhui et al., 2009) it was presented a system that used intertial and electromyographic sensors worn by the user, and a camera in the environment to recognize the dynamic gestures performed by the user. This implementation of the pervasive computing paradigm required the development of advanced data fusion techniques. The sensors fusion allowed to achieve gesture recognition accuracies that are quite higher than the accuracies obtained using a single sensor type.

All the aforementioned systems presented in the last subsection show that a proper combination of the wearable and environmental paradigms allow to develop a system that can alleviate weaknesses due ot the adoption of a single paradigm. A system that is developed following the pervasive computing paradigms is composed of a wearable subsystem and an environmental subsystem. However, all the systems presented in this subsection need the simultaneous presence and functioning of the wearable and environmental subsystems limiting the user's freedom and lacking of computational optimization. Indeed, in the discussed examples of pervasive computing systems if one of the two subsystems stops working, the functioning of the whole system is compromised. The pervasive computing paradigm that requires the simultaneous functioning of the wearable and environmental systems is called "complementary type". Recent studies tried to introduce smarter architectures for the combination of the wearable and environmental paradigms that do not oblige the system to have a static composition. Roggen et al. presented an opportunistic paradigm for activity recognition that leavarages a system capable of optimizing the recognition methods in order to dynamically adapt to the available sensors data (Roggen et al., 2013). That allows creating reliable gesture and activity recognition applications despite the changing sensors availability. In this thesis, a novel paradigm called "synergistic paradigm" is presented to push further this concept. This paradigm allows developing a system that can dynamically recognize gestures depending on the availability of the wearable and

environmental subsystems. The wearable and environmental subsystems can function independently but if combined they use a fusion engine, which allows increasing the gesture recognition accuracy. The opportunistic paradigm leverages a system that profits of the advantages coming from both the wearable and environmental subsystems combined and at the same time grants the gesture recognition accuracy being no lower than the best accuracy obtained with the single subsystem.

## 2.5 Gesture recognition in smart environments

In this scenario of ubiquitous computing, the sensors are distributed in both on the user and in the environment. The sensors on the user are supposed to be part of a personal system that dynamically improves in recognizing the gestures performed by the user; on the other hand, the sensors in the environment have to adapt to many different users and to different conditions. Most of these environmental systems are based on vision-based technology. The vision-based gesture recognition systems have to face with a particular problem, which consists in the user's freedom of body movements. In particular, the user can assume different positions in the environment while gesturing and this factor can severely compromise the recognition process. Indeed, some works focused on developing vision-based systems for view-invariant gesture recognition. In this case, the information provided by a single 2D camera is not adequate for accurate 3D gesture recognition since it makes the system strongly dependent on he specific viewpoint. Nevertheless, the combination of multiple RGB cameras from different viewpoints allows the reconstruction of 3D information facilitating 3D gesture recognition. For example, in (Peng et al., 2009) the voxel data extracted from the different silhouettes coming from several RGB cameras allow the 3D reconstruction of the user's body. After a phase of post-processing of these data, they used multilinear analysis for the posture recognition and the HMM for the gesture recognition. Another solution based on RGB cameras was presented in (Roh et al., 2006) where the authors used only two 2D cameras in

their gesture recognition system. Using the disparity map calculated on the images provided by the cameras and a particular appearance-based method, they were able to extract the volume motion template, which allowed estimating the optimal virtual view-point and then classifying the data using the least square method and k -nearest-neighbor algorithm. Unfortunately, RGB cameras are not resilient to illumination changing and they limit the application of these systems in controlled environments. Yuan et al. avoided the lighting changing and the hand-tracking problems using two infrared cameras with IR LEDs and a retro-reflective marker (Yuan et al., 2010). In this work, the system was able to recognize only 3D trajectories of non-directional gestures that approximately lie on a plane neglecting the rest of dynamic gestures. Another strong disadvantage related to this approach was the intrusiveness due to the need of putting markers on the user; in fact, this constraint should imply the classification of this system in the complementary type of the pervasive paradigm. Another interesting approach was presented in (Holte et al., 2010), where the authors used a 3D camera, in particular, a time-of-flight camera. They used the intensity and range images provided by the camera to calculate 3D motion primitives of the user's body movements. The tests demonstrated that using this approach it is possible to recognize four arm-gestures with good accuracy allowing the user to freely change position (i.e., varying the user's angulation with reference to the optical axis of the camera lens) between -45° and +45°. The use of a single 3D camera reduces the amount of data that can be captured to reconstruct the 3D information. In fact, using two 3D cameras could allow expanding the interaction are and the angulation range enabling the user to perform dynamic gestures more freely. On the other hand, using multiple 3D cameras introduces a new challenge, which consists of the calibration between the different cameras. In this thesis, it will be provided a calibration procedure for multiple 3D cameras and two algorithms for the 3D gesture recognition of deictic and dynamic gestures.

## 2.6 Gestures can be ambiguous

Gestures represent a very important modality for the human communication. Unfortunately, as sometimes it happens in the verbal communication, gestures can be misunderstood. Gestures are difficult to characterize. Generally, there is not a direct, one-to- one mapping between a gesture and a meaning and such association is strictly bound to the performer experience, language and culture (Mitra & Acharaya, 2007; Norman, 2010). The same gesture can be executed differently by different users and even by the same individual if performed in different contexts. Therefore, gestures can be ambiguous and often the context is the key for a correct interpretation of the performed signs (Wachs et al., 2011). The following paragraph reports some examples in order to make clear how gestural communication can be ambiguous if not related to the specific context.

### 2.6.1 The V-sign gesture

The V sign is a hand gesture in which the index and middle fingers are raised and parted to form a V, while the other fingers are clenched (Figure 2.4). This particular hand gesture can have many different meanings. One of the most common meanings associated to this gesture is the representation of the number two for quantity. In particular, the American Sign Language (ASL) includes this gesture for the symbolic embodiment of the quantity two (Liddell, 2003). Although it is quite simple to understand this gesture, it is possible to represent the number two with other hand postures depending on the cultural context. Another very popular meaning associated to this gesture is victory. The “etymology” of this meaning dates back to the period of the World War II. On January 14, 1941, the director of the Belgian French version of BBC, the former Belgian Minister of Justice Victor de Laveleye, suggested in a broadcast that Belgians should use a V for victory (victoire in French and vrijheid in Dutch) as an emblem against the occupier. Victor de Laveleye stated that "the occupier, by seeing this sign, always the same, infinitely repeated, [would] understand that he is surrounded, encircled by an immense crowd of citizens eagerly awaiting his

first moment of weakness, watching for his first failure." Within weeks chalked up Vs began appearing on walls throughout Belgium, the Netherlands, and northern France. Buoyed by this success, the British BBC started the "V for Victory" campaign. Many forms of communication of the V has been adopted such as a special symbol in Morse code (as a three dots and a dash), on posters, on stamps and as gesture. In fact, the V was also represented through the V sign as a greeting gesture. One of the greatest supporter of the “V for Victory” campaign was the British Prime Minister Winston Churchil, who referred approvingly to this campaign in a speech. From that point he started using the V hand sign and many diffused photographs depict him performing the V sign gesture. Other allied leaders followed his example and adopted this greeting gesture to support this campaign as a psychological war against the occupiers. The victory meaning still remains associated to this gesture but always in the context of this same gesture can have another meaning: peace. During the Vietnam War, the U.S. president Richard Nikon used this very gesture to signal the American victory on the communist allies. Protesters against the Vietnam War and counterculture activists adopted the V sign gesture as a sign of peace. Because the hippies of the day often flashed this sign (palm out) while saying "Peace", it became popularly known (through association) as the peace sign. A very famous example is the artist and activist John Lennon, who has often been portrayed performing this gesture. This meaning spread all over the world also thanks to the evolution of the communication systems, and it changed in some peculiar cultural contexts. For instance, V sign is very commonly made by Japanese people, especially younger people, when posing for informal photographs. The reason of this practice can be attributed to the American figure skater Janet Lynn. During the 1972 Winter Olympics in Sapporo, She fell during a free-skate period, but continued to smile even as she sat on the ice. Though she placed third in the competition, her cheerful diligence and persistence resonated with many Japanese viewers. Lynn became an overnight foreign celebrity in Japan. Since she was also a peace activist, she was often portrayed performing

the V sign (meaning peace) in the Japanese media. Probably, the emulation of the foreign star by the younger people made this practice become so successful. This gesture, in the consolidated usage in the photo posing found also another meaning as “rabbit ears”. This gesture has to be meant as a joke to do with friends in photos. In this case, the V sign is performed behind the friend’s head in order to recall the long rabbit’s ears.

When describing the V sign, no rule exists about the hand orientation. Funnily, it happens that performing the V sign with the pal inwards (that means facing the gesturer) is considered an insult in some countries such as Australia, New Zealand, South Africa, Ireland, and the United Kingdom (Victor & Dalzell, 2007). This gesture is called "two-fingered salute", or "The Longbowman Salute", or "the two", or "The Agincourt Salute". The reason of these epithets is due to the controversial and uncertain origin of this meaning. The most famous theory attributes the invention of this gesture as an insult to the English and Welsh armies of the 15$^{th}$ century. In particular, legend claims that the two-fingered salute or V sign derives from a gesture made by longbowmen fighting in the English and Welsh army at the Battle of Agincourt during the Hundred Years' War. According to the story, the French were in the habit of cutting off the arrow-shooting fingers of captured English and Welsh longbowmen, and the gesture was a sign of defiance on the part of the bowmen, showing the enemy that they still had their fingers.

**Figure 2.4 V sign hand gesture.**

Performing the V handshape can also mean the letter V in ASL during spelling (Klima, 1979). However, it is sufficient to perform the V sign with the thumb not very clenched to the hand and its meaning can vary. In fact, if the V handshape is produced with the index and middle fingers extended, and with the thumb in contact with the middle finger, this gesture (also called chopstick hand) means the letter K always in ASL (Klima, 1979).

While playing the games “Rock, paper scissors” and “Rock-paper-scissors-lizard-Spock”, the V sign (with a different orientation, usually with the index and middle fingers parallel to the floor) represents the scissors.

The V sign is a hand posture, but to add a movement can again radically its meaning. Performing the V sign as to represent scissors in “Rock, paper scissors” moving the index and middle fingers to approach and then to part again can mean cutting (because it should recall the movement of cutting scissors) also out of the game context. A different movement can completely transform the message carried by the gesture. In fact, performing the V sign with the fingertips oriented down and with the hand making a smooth horizontal movement from right to left as the fingers wiggle, followed by a hold during which the fingers do

not wiggle means that a person is walking from a place to another. This gesture in ASL is known as the verb BIPED-WALK-TO (Liddell, 2003).

### 2.6.2 The A-ok gesture

The V sign is a very popular gesture but it is not the only one that can be easily misunderstood. Another important example is the A-ok hand gesture. In this gesture the tip of the index finger is brought into contact with that of the thumb, so that there is a circular gap between them, and holding the other fingers straight or relaxed in the air (Figure 2.5). The meaning that gives the name to this gesture is OK transmitting the messages: “it is OK”, or “I am OK”, or “Are you OK?.” However, this gesture assumes other meaning in other social or cultural contexts. For instance, this gesture is called “Vitarka mudra” and is used as the gesture of discussion and transmission of Buddhist teaching (Hirschi, 2000). In fact, with this gesture the deity or Buddha underlines the meaning of the words. A similar concept is applied in southern Italy, where this gesture is called The Ring. The Ring is an Italian gesture used in conversation to delineate precise information, or emphasize a specific point. It is made similarly to the A-ok sign, but the ring made by the thumb and forefinger is on top with the palm facing medially. The arm moves up and down at the elbow. If more emphasis is needed both hands will make the gesture simultaneously with the palms facing one another (Kendon, 1995).

**Figure 2.5 The A-ok hand gesture.**

Although, in western culture the A-ok gesture has a positive valence and indicates approval, in other cultures can have the opposite meaning. For instance, in the Arab world if the gesture is shaken at another person it symbolizes the sign of the evil eye. An Arab may use the sign in conjunction with verbal curses as threatening gesture (LaFrance & Mayo, 1978). This association to an evil sign is also known in other cultural contexts, where the circle with the middle finger stands for a 6, the three fingers raised represent three sixes, or 666. In some countries (e.g., Brazil, Somalia and Southern Europe), this sign is an insult or an obscene gesture even if it is not related to the occultism (Skelton & Cooper, 2004; Hendon et al., 1996; Bolton, 2001). Another completely different meaning for the A-ok can be found in Japan, where it means money when used with the back of the hand facing down and the circle facing forward.

Another popular meaning associated to the A-ok gesture is zero, since the circle formed by the thumb and the index finger recalls the round shape of that number in the Arabic numerals. However, the same gesture represents the number nine in the ASL (Klima, 1979). A particular attention must be paid to it by a gesturer while performing this gesture using the ASL because this gesture, when made with the thumb and forefinger parallel to the ground, is an insult.

As already explained, the A-ok gesture signs the number nine in the ASL. Usually, the finger counting to nine does not include such a gesture. In fact, also the gesturing in finger counting, or dactylonomy, change with reference to the ethnographic context (Pika et al., 2009). For English speakers, primarily in North America and the United Kingdom, and occasionally in Australia, the count to 5 starts with the extension of the index finger (number 1) and continues to the little finger (number 4). The extension of the thumb indicates five. The process is repeated on the other hand for numbers up to 10. For Western Europeans, such as Germans, Italians, Belgians, Austrians, the Swiss, the Dutch, the Spanish, or the French, the thumb represents the first digit to be counted (number 1). The index finger is number 2 through to the little finger as number 5. Fingers are generally extended while counting, beginning at the thumb and finishing at the little finger. Eastern Europeans generally use the same system as Western Europeans. However, for Russians and citizens of former USSR countries, counting begins with all digits extended. Numbers are expressed by folding fingers and the thumb inwards. Finger counting systems in use in many regions of Asia allow the counting to 12 by using a single hand. The thumb acts as a pointer touching the three finger bones of each finger in turn, starting with the outermost bone of the little finger. One hand is used to count numbers up to 12. The other hand is used to display the number of completed base-12s. This continues until twelve dozen is reached, therefore 144 is counted. In Japan counting for oneself begins with the palm of one hand open. Like in Eastern Europe, the thumb represents number 1; the little finger is number 5. Digits are folded inwards while counting, starting with the thumb. A closed palm indicates number 5. By reversing the action, number 6 is indicated by an extended little finger. A return to an open palm signals the number 10. However to indicate numerals to others, the hand is used in the same manner as an English speaker. The index finger becomes number 1; the thumb now represents number 5.

### 2.6.3 The role of context

The previous subsections show some examples of how important context is when decoding a gesture. In fact, one gesture can have several different meanings but it can happen that different gestures can have the same meaning in some contexts. Hence, gestures are ambiguous and incompletely specified. For example, to indicate the concept “stop,” one can use gestures such as a raised hand with palm facing forward, or, an exaggerated waving of both hands over the head. They are very different gestures: the first is static (hand posture) and the second one is dynamic. The use of these different gestures depends of the context: if the gesturer is near to the message receiver, then he will preferably raise his hand in the aforementioned posture to mean “stop”. On the contrary, if the gesturer is far from the message receiver, then he will augment his gestural communication visibility preferring the second gesture.

Hence, gestures are incompletely specified and, in addition, they are ambiguous. In (Mitra & Acharya, 2007), the authors stated that the meaning of a gesture is mainly dependent on four parameters:

- Spatio-temporal information: the context of where and when a gesture occurs. For instance, gestures performed in kitchen while cooking or in the living room while watching a movie can assume different meanings.
- Pathic information: in hand-gestures above all, the path described by the hand movement is the most informative source of features, especially for dynamic gestures.
- Symbolic information: due to the cultural context, gestures are associated with symbols. Therefore, the mapping between the symbol and the meaning can be straightforward but depending on the cultural context.
- Affective information: humans feel and react to the surrounding ambient and experience with emotions using also their own body and gestures. Social dynamics (and genetics) make them to react in similar manners to

the same emotional stimuli. In this way, specific gestures can be associated to a well-defined emotion.

Therefore, an intelligent system should be able to model the information belonging to these four parameters in order to correctly interpret the meaning of a specific gesture.

## 2.7 Context-aware gesture recognition

Gestures can be ambiguous and many researches suggest using context as the key to interpret their meaning (Wachs et al., 2011; Mitra & Acharya, 2007). The problem lies in defining what context is.

### 2.7.1 Definition of context

A first definition can be found in the Oxford English Dictionary, which defines context as:

The whole structure of a connected passage regarded in its bearing upon any of the parts which constitute it; the parts which immediately precede or follow any particular passage or 'text' and determine its meaning.

This definition is very near to the context with reference to the spoken or written language. This definition depicts the complexity of context and the fact that is structured. The parts of the structure can provide different kinds of information to understand the meaning of a passage or text. Since this definition comes from a dictionary, this description is quite general. The word context can assume very different meaning with reference to the concerned domain. For instance, in Psychology context refers to the background stimuli that accompany some kind of foreground event. For example, if a rat is foraging and is frightened by a cat, the place (and possibly time) of foraging is the context and the cat is the foreground event. There seems to be a specialized neural structure, the hippocampus, for the processing of some kinds of context. In Philosophy, the

context gave the rise to a trend called Contextualism, which describes a collection of views in philosophy which emphasize the context in which an action, utterance, or expression occurs, and argues that, in some important respect, the action, utterance, or expression can only be understood relative to that context (Price, 2008). The same happened in art, where a German trend took that name of Context Art (Kontext Kunst in German). About Context Art, Peter Weibel wrote in (Weibel, 1994):

"It is no longer purely about critiquing the art system, but about critiquing reality and analyzing and creating social processes. In the '90s, non-art contexts are being increasingly drawn into the art discourse. Artists are becoming autonomous agents of social processes, partisans of the real. The interaction between artists and social situations, between art and non-art contexts has led to a new art form, where both are folded together: Context art. The aim of this social construction of art is to take part in the social construction of reality."

Context became a main topic also in computer science. Already back in the 1960s. the notion of context has been modeled and exploited in many areas of computer science (Coutaz et al., 2005). The scientific community has debated definitions and uses for many years without reaching a clear consensus (Dourish, 2004). Schilit et al. provided a definition in order to adapt the notion of context-aware systems to the emerging mobile computing (Schilit et al., 1994):

"Context encompasses more than just the user's location, because other things of interest are also mobile and changing. Context includes lighting, noise level, network connectivity, communication costs, communication bandwidth and even the social situation, e.g., whether you are with your manager or with a co-worker."

In 2001, Dey and Abowd gave a definition of context where they introduced the concept of entities characterized by individual states (Dey et al., 2001):

“Context: any information that can be used to characterize the situation of entities (i.e. whether a person, place or object) that are considered relevant to the interaction between a user and an application, including the user and the applications themselves. Context is typically the location, identity and state of people, groups and computational and physical objects.”

In the same year Moran and Dourish described context as physical and social situation in which computational devices are embedded (Moran & Dourish, 2001). This extension was important because it specified that the state is more than the physical status but there are many other factors linked to cultural and social conditions that can determine the meaning of an action. The most comprehensive definition of context has been given by Zimmermann et al., who wrote in (Zimmermann et al., 2007):

“Context is any information that can be used to characterize the situation of an entity. Elements for the description of this context information fall into five categories: individuality, activity, location, time, and relations. The activity predominantly determines the relevancy of context elements in specific situations, and the location and time primarily drive the creation of relations between entities and enable the exchange of context information among entities”.

This operational definition introduces the five classes that can be used to categorize the contextual information retrieved by systems. Moreover, the time category indicates how important is the description of the different entities at a certain time. In fact, context is not simply the state of a predefined entity with a fixed set of parameters. It is part of a dynamic process of interacting between the different entities with an ever-changing environment composed of reconfigurable, migratory, distributed and multi-scale resources (Coutaz et al., 2005). The aspect of dynamism of context is at the base of the distinction between the two major approaches to context information management. The two approaches for context modeling in human-computer interaction are:

positivism and phenomenology (Dourish, 2004). Positivist design presents context as a set of features of the environment surrounding generic activities. These features are retrieved through sensors distributed in the environment, and the data are encoded and made available to a software system that can use this information as enclosure for the reasoning about the activity. With reference to this approach, context is a form of information that can be measured and represented in software. Moreover, it is possible to define what parameters count for the determination of the context in specific activities defining in advance the functions that the system supports. This leads to another relevant characteristic of the positivist approach: context and activity are considered two separate elements. The activity is a set of actions performed in a context and the context describes a set of features characterizing the environment but that are separate from the activity itself. Moreover, this set of features that describes the context is stable. Although the precise elements of a context representation might vary from application to application, they do not vary from instance to instance of an activity or an event. The determination of the relevance of any potential contextual element can be made once and for all. On the contrary, the phenomenological approach defines the context as an interactional problem, which is dynamic and its features cannot be defined in advance. In fact, the context is particular to each instance of an activity. The context is not only considered as information to be measure but is represented as a relational property that holds between objects or activities. It is not simply the case that something is or is not context; rather, it may or may not be contextually relevant to some particular activity. Finally, context and activity are not separable: context arises from the activity.

### 2.7.2 Context models in ubiquitous computing

A definitive model for context information has not been created yet but the research community, in particular in the pervasive computing domain, is looking for an adequate context information modeling and reasoning techniques that could be used in context-aware applications. In fact, these techniques reduce the

complexity of context-aware applications and improve their maintainability and evolvability. Bettini et al. described the seven fundamental requirements defined for the context modeling, management and reasoning (Bettini et al., 2010):

Heterogeneity and mobility: context information can come from very different sources and a good model should take it into account.

Relationships and dependencies: context is also described by the various relationships between types of context information that have to be captured to ensure correct behavior of the applications.

Timeliness: the time reference for the retrieval of information is crucial, also the possibility of retrieving past data can add very important information to the current context.

Imperfection: a good model should take into account that context information can be of variable quality because it is dynamic and heterogeneous.

Reasoning: the system should be able to adapt and to take a decision whether it is necessary despite the changing context.

Usability of modeling formalisms: the models should be simple and easily understandable by designers and developers in order to allow them to facilitate their work.

Efficient context provisioning: efficient access to context information is needed, which can be not so easy in presence of big amount of data.

Different approaches have been suggested to develop context-aware applications. Dey et al. have proposed a rapid prototyping framework (Dey et al., 2001). That model inspired many following works but it did not take into account the central role of the people tasks and activities, which have been introduced by different researchers later (Crowley et al., 2002; Kofod-Petersen & Cassens, 2005; Chen et al., 2004; Henricksen & Indulska, 2003). Since the context information is extremely dynamic, (Hynes et al., 2009) introduced a context life-

cycle model to manage the information using web-services; that work has been extended and improved in (Villegas & Müller, 2010). All these works model the contextual information on low levels. Hong et al. made a literature review and presented a model to represent the different layers that compose a context-aware system (Hong et al., 2009). At the lowest level there is the network Infrastructure Layer; above that there is the middleware layer and then the Application Layer. On top of them all there is the User Infrastructure Layer, where the user interface is. Figure 2.6 depicts the different layer, image taken from (Hong et al., 2009). For the top layer, few works exist in the literature.

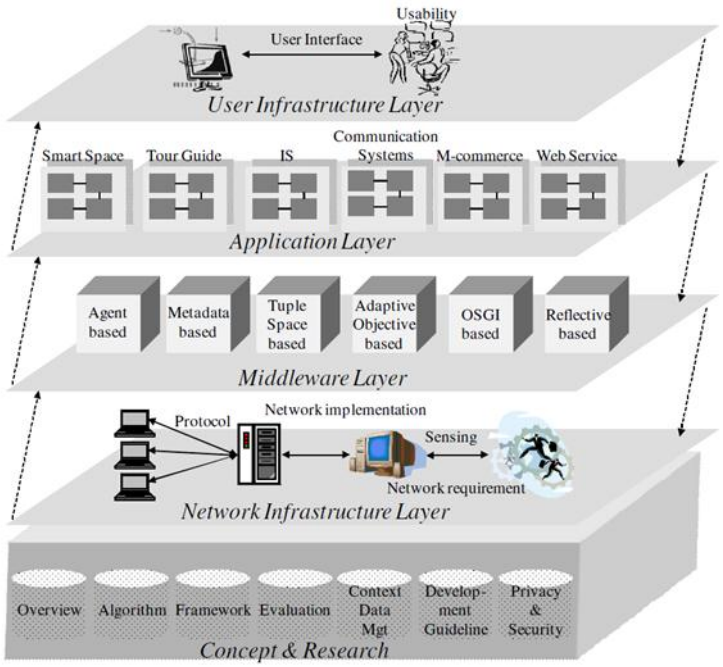


**Figure 2.6 Classification framework of context-aware systems (Hong et al., 2009).**

A very interesting work has been presented in (Widjaja & Balbo, 2005), where the authors designed the roles that the context awareness can assume during the user interaction. They proposed three high-level spheres of role based on the extent of how the context is used to influence the interaction, namely interpretation, representation, and effectuation. Figure 2.7 depicts these spheres; the image has been taken from (Widjaja & Balbo, 2005).

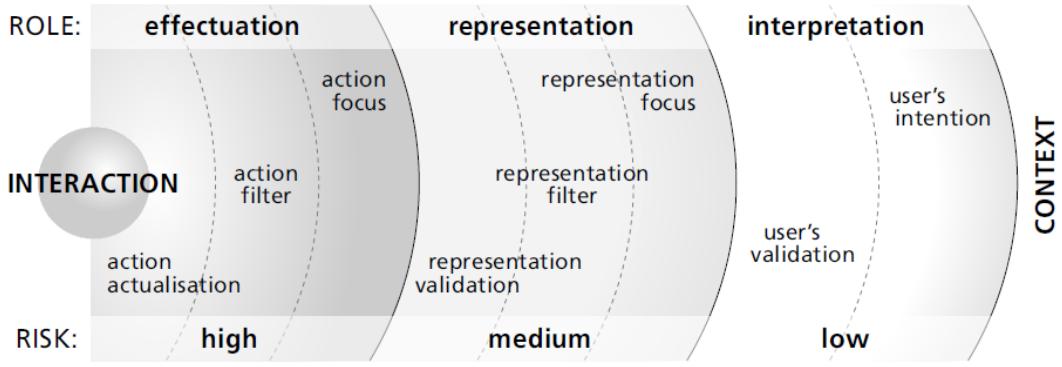


**Figure 2.7 Spheres of role of context awareness in the user interaction (Widjaja & Balbo, 2005).**

Roles in Interpretation Sphere: Interpretation takes context into the understanding of an activity's meaning. This interpretative role may be elaborated as follows:

- Interpreting user's intentional action
- Interpreting user's validity for an action

Roles in Representation Sphere: The role of context in this aspect is about providing the appropriate and optimal representation to the user.

- Providing representation focus
- Reducing representation amount
- Providing valid representation

Roles in Effectuation Sphere: Effectuation provides further utilization of contextual data.

- Suggesting appropriate action
- Reducing action space
- Acting on user's behalf

Applying the context awareness to a system that provides natural interaction through gestures can add a new role to the Effectuation Sphere: disambiguating the command, i.e., the gesture meaning. Applying these principles in the gestural interaction domain could be interesting and some researchers already stressed

the importance of developing a framework in this new scenario of context-aware interaction in smart environments (Selker & Burleson, 2000). Unfortunately, only few researches about frameworks for context-aware gesture recognition exist. Some examples about context-aware gestural interaction can be found in the literature. For instance, (Raffa et al., 2010) developed a system that recognizes when the user is or is not performing a command gesture to optimize battery consumption; (Oh et al., 2011) use a 3D accelerometer on a 2D space to recognize user's activity through gestures and context information (e.g., GPS coordinates) using Parametric Hidden Markov Model (PHMM); (Paulson et al., 2011) used hand postures to determine the types of interactions a user has with objects in a desk/office environment; Sato et al. (2007) developed a system composed of three unconscious robots to recognize pointing gesture and two visible-type robots provided service to the user; the Medusa project, augmented the gesture recognition of a tabletop thanks to the information related to the arm and to the user (Annett et al., 2011). All the aforementioned examples explored the context-aware gestural interaction but they were very specialized in one application. What is really missing is a generic framework for the use of context information to improve the interaction. This framework should be generic in order to be applied to different scenarios and to enable practitioners and researchers in the human-computer interaction to start creating a more methodic process of design for the gestural interfaces. Unfortunately, as stated by Norman and Nielsen, very few works focused on this topic. A first example of an effort in such direction can be found in Karam's work, in particular about the gesture classification, which has been previously described. This is a very preliminary work but framework should allow more concrete guidelines for the design and development of gestural interfaces. In fact, the gesture classification, even if is very important, does not provide a real tool to enhance this process. An example of guideline for the development of gestural interfaces for the interaction with a smart environment can be found in (Stocklöw & Wichert, 2012); they proposed an architecture and an ontology for the development

systems that can recognize gestures and associate a semantic service. This work proposes an interesting gestural interaction with the environment but, unfortunately, their proposal does not abstract the elaboration of the contextual information and is strictly development oriented. In fact, this work takes into account only the distance gesture interaction and is conceived for the only application to the ambient assisted living. Greenberg et al. proposed a good example of real framework for context-aware gestural interfaces (Greenberg et al., 2011); this model is based on the spatial relationship between the users and interactive displays. In this scenario, users can interact with many digital surfaces and portable personal devices; this model introduces five dimensions to characterize the entities and to describe the interaction in order to improve the user experience. Unfortunately, this framework has been conceived to describe the interaction with only touch-enabled surfaces and portable personal devices; this choice represents an important issue in the scenario of the ubiquitous computing. In fact, Greenberg et al.' framework does not take into account the gestural interaction with objects (tangible interaction) and, in particular, for gestures performed in the air.

The literature review shows that there is a lack of a high-level framework for the organization of the contextual information for gestural interfaces of the ubiquitous computing era; in this era, the interaction goes beyond the screens and the user should be able to interact with the entire surrounding environment using different kinds of gestures. The context awareness could improve the interaction helping in many ways; the most important maybe it is disambiguating the meaning of a gesture. A high-level framework for context-aware gestural interfaces should be applicable to all the systems integrated in the environments of the ubiquitous computing era and should provide the tools to optimize the gesture design.

## 2.8 Summary

This chapter presented the literature review concerning the gestural interaction in smart environments. Section 2.1 introduced the importance of this communication modality, which was and still is fundamental for the human beings to interact with the surrounding environment. Section 2.2 and Section 2.3 presented the different categories for gesture classification, in both psychological and computer science domains; it is highlighted that the most important categories for gesture classification in HCI derived from the psychological researches. Although these different categories are perfectly suitable to differentiate the different kinds of gestures that exist, they do not provide a method to design gestures for "natural" gestural interfaces. Some researches coped with this issue and presented the main characteristics that gestural interface should have in order to be as "natural" as possible for the users. Other researchers presented some methods that can help to design the gestures in order to enable designers and developers to implement a "natural" gestural interface; in the subsection 2.3.3, some examples such as the role-playing, Wizard of Oz and self-definition are reported. These techniques can be used as tools to facilitate the gesture design but they have been conceived to associate a gesture to a single function. Mapping one gesture to a single function can be a problem, in particular in the current interaction scenarios, where the smart environments are designed to provide many different services. In fact, mapping one gesture to one function means creating a gesture taxonomy that is very vast if the smart environment is able to provide many functions. A vast taxonomy is a problem for the user since it reduces its usability and learnability; these two are among the most important characteristics identified by the researchers that should characterize a "natural" gestural interface. Therefore, a novel method for the gesture-to-function mapping is presented in this thesis. This method allows reducing the number of different gestures to be associated to the functions in order to create gesture taxonomies that can be easily learnt

by the users. This method involves the implementation of a framework for the context-aware gesture recognition.

Section 2.4 analyzed the concept of ubiquitous computing, which is the fundamental principle that is inspiring the current research in computer science. The ubiquitous computing is the general frame where the modern research about gestural interfaces is situated in. In particular, three different paradigms were identified for the design and development of gestural interfaces: environmental, wearable and pervasive. These paradigms have different advantages and disadvantages; in this thesis, a particular type of pervasive paradigm is presented, which aims at opportunely mixing the wearable and environmental paradigms in order to merge the advantages of these paradigms. This paradigm is applied to the gesture recognition in smart environments of the ubiquitous computing era.

Section 2.5 presents the different solutions for the dynamic gesture recognition in smart environments that are present in the scientific literature. The literature shows that the most used approach for the dynamic gesture recognition in smart environments is vision-based. Unfortunately, this technology has a particular problem, which is the dependency between the user position with reference to the camera and the recognition accuracy. Usually, the user position severely affects the system performance reducing the gesture recognition accuracy. A smart environment should enable the user to interact with full freedom of movement and for this reason some researchers presented some solution in order to develop a technique that can leverage a view-independent gesture recognition system. Some solutions are interesting but the best performance is provided in (Holte et al., 2010), where the authors developed a technique that allows recognizing four arm-gestures with good accuracy allowing the user to freely change position (i.e., varying the user's angulation with reference to the optical axis of the camera lens) between -45° and +45°. In this thesis, a novel techniques based on two calibrated depth cameras is presented in Chapter 4;

this technique allows recognizing deictic and dynamic gestures with an augmented freedom of movement comprised between -90° and +90°.

Section 2.6 reports two examples to show the relationship between the gesture meaning and the context. The role of context is fundamental to decode the meaning of a specific gesture in the human society; this is also true in the interaction with smart environments. Indeed, Section 2.7 defines the meaning of context in computer science and presents the models present in literature to develop context-aware systems. This analysis focuses on the frameworks for the development of context-aware gestural interfaces. In particular, Greenberg et al.'s work addresses this issue providing a model to describe the spatial relationship between the different entities that are present in the same room (i.e., users and interactive screens) as shown in (Greenberg et al., 2011). Althoguh this approach is very effective for the development of context-aware touch-enabled surfaces, it does not take into account a distance interaction with gestures performed in the air. Since there is a lack of a high-level framework that can be adapted to all different types of gestural interaction (i.e., touch gestures, tangible gestures and gestures performed in the air), this thesis presents a novel framework for the modeling of gestural interactions in smart environments of the ubiquitous computing era. This framework aims at facilitating the development of gestural interfaces with a special regard to the optimization of the gestures taxonomies as it will be further explained in Chapter 3.

# Chapter 3: Context-Aware Gestural Interaction

## 3.1 Introduction

The literature review presented the works that treated the topic of gestural interaction in smart environments and this analysis highlighted that in the literature there is a lack of a high-level framework that can facilitating the development of gestural interfaces with a special regard to the optimization of the gestures taxonomy design. Moreover, this framework should be high-level because it should be adaptable to all different types of gestural interaction (i.e., touch gestures, tangible gestures and gestures performed in the air). In this thesis, a novel high-level framework for context-aware gestural interaction is presented. It is expressly conceived to include all different types of gestural interaction that can be performed in a smart environment. Moreover, it introduces the concept of "functional gesture", which allows designing and developing user interfaces with a taxonomy that is able to reduce the user's cognitive load and, at the same time, to augment the learnability. The functional gesture concept ultimately aims at providing a higher usability of the gestural interface to improve the user experience in the smart environments of the ubiquitous computing era.

This framework is not limited to describe the interaction between the user and the smart environment but it involves also the concepts of user's status and system status (i.e., the smart environments as composed of multiple smart objects).

## 3.2 A framework for context-aware gestural interaction

The literature review showed how the context awareness is treated in the development of a system in the ubiquitous computing era. In particular, the application of the contextual information changes with reference to the system layer. In the human-computer interaction, the context-aware services are of great value because they can enhance the user experience in many aspects. Using this information can be treacherous and a framework could improve the computer scientists working in the HCI field to create new systems that can better exploit the contextual information to improve the user interface. As first step, it was necessary to choose the best approach for the representation of the contextual information. The four standard approaches for context modeling and reasoning that are the protagonists of the current state-of-the-art:

- Object-Role Modeling, in particular Context Modeling Language
- Spatial models
- Ontology-based model
- Hybrid approach

Since the goal was to create a framework for the design and development of an interface for the human-smart environment interaction, the spatial model was the obvious choice. In fact, the spatial reference plays a key role in the interaction between humans and the environment. The context information can be gathered from the spatial representation of the entities that are present in the environment. The entities are divided in two types: the humans and the smart objects. The spatial relationship between these entities can characterize the interaction. The spatial model is composed of different levels and these levels can provide different information about the five categories that describe the entities and their contexts. The scalability of space model can be expressed as geographical position and relative location:

- Geographical position: country (social-ethnographic information, if in his country with historical data, if in holyday, hour).
- Geographical position: city (information about transferring from one city to another: business travel, holidays, home et cetera).

- Geographical position: specific building (at work, at home, at gym et cetera).
- Relative location: specific room referring to the specific building (kitchen, living room, bedroom et cetera).
- Relative location: coordinates referring to the specific room.

The spatial relationship between the different entities that are present in the same room refers to the concept of proxemics, which derives from the psychology; E. T. Hall gave the definition of proxemics as "the study of how man unconsciously structures microspace—the distance between men in the conduct of daily transactions, the organization of space in his houses and buildings, and ultimately the layout of his towns" (Hall, 1963). In (Vogel & Balakrishnan, 2004), the authors elaborated this concept applying it to the public displays. They took Hall's interpersonal interaction zones: intimate (less than 1.5 feet), personal (1.5 to 4 feet), social (4 to 12 feet), and public (12 to 25 feet). Then, they characterized the human-display interaction in the same manner. Some years later, Greenberg et al. presented a model based on proxemics theory for the interaction between people and displays, extending Vogel and Balakrishnan's work (Greenberg et al., 2011). The difference is that in this proxemics ecology, the displays are composed of digital surfaces, portable personal devices and information appliances. This scenario is much more similar to Weiser's scenario of ubiquitous computing interface. In order to model the proxemics for this ubiquitous computing interface, they introduced five dimensions to characterize the entities and to describe the interaction. These dimensions are: distance, orientation, movement, identity and location. However, this scenario excluded the possibility of interaction through air gestures and tangible gestures. Moreover, some of these measures are not actually sensed but they can be calculated from the other measures. For instance, the distance measure can be calculated using the positions of the different entities. For this reason the dimensions acquired from the sensors can be transformed in: orientation, movement, identity, location, time and state. The orientation and location describe the positions of the different entities in the environment. The identity

and the status dimensions provide some information about the characterization of the specific entities. The time is important especially for the construction of a history of the interaction and for the habits. The movement dimension refers to the Oxford English Dictionary definition: “The power or facility of voluntary movement of a part of the body.” In particular, this dimension represents the movements performed by the user, i.e., gestures, or the moving parts of a smart object.

These dimensions represent the data acquired by the sensors and they must be elaborated to obtain a higher level of the understanding of the context. The different levels of understanding can be classified as in the Data Information Knowledge Wisdom (DIKW) pyramid (Rowley, 2007), see Figure 3.1.

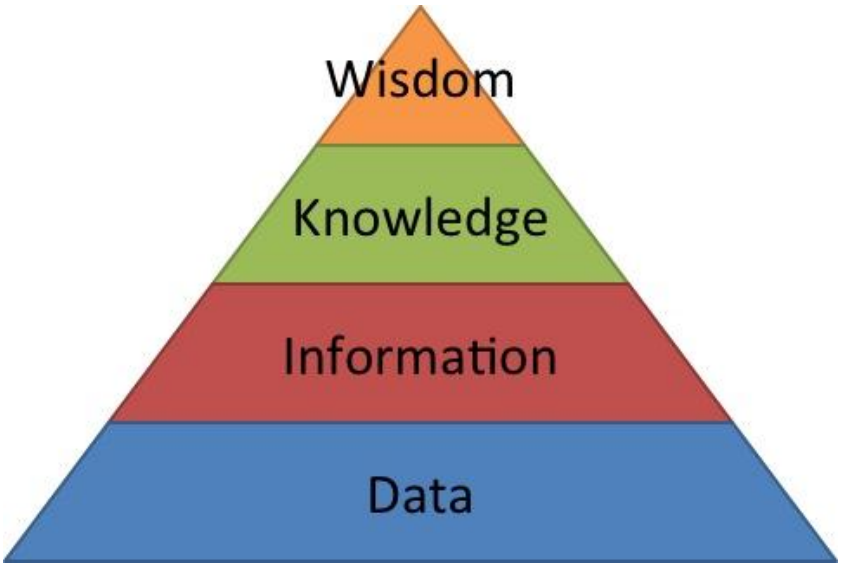


**Figure 3.1 The Data Information Knowledge Wisdom (DIKW) pyramid.**

The data are raw; they represent facts and have no significance beyond their existence. The information is elaborated data: the data have been given meaning by way of relational connection and the information embodies this relationship. The knowledge is the appropriate collection of information, such that its intent is to be useful. The information has been elaborated in order to find patterns, which provide a high level of predictability as to what is described or what will happen next. The wisdom is more complex. It is an extrapolative process and it calls upon all the previous levels of knowledge and consciousness. Wisdom

embodies more of an understanding of fundamental principles embodied within the knowledge that are essentially the basis for the knowledge being what it is. Bellinger et al. gave this description and the following graph is their graphical interpretation of the links between the different layers (Bellinger et al., 2004). Figure 3.2 represents the different levels of information elaboration associated to the DIKW layers, image adapted from (Bellinger et al., 2004).

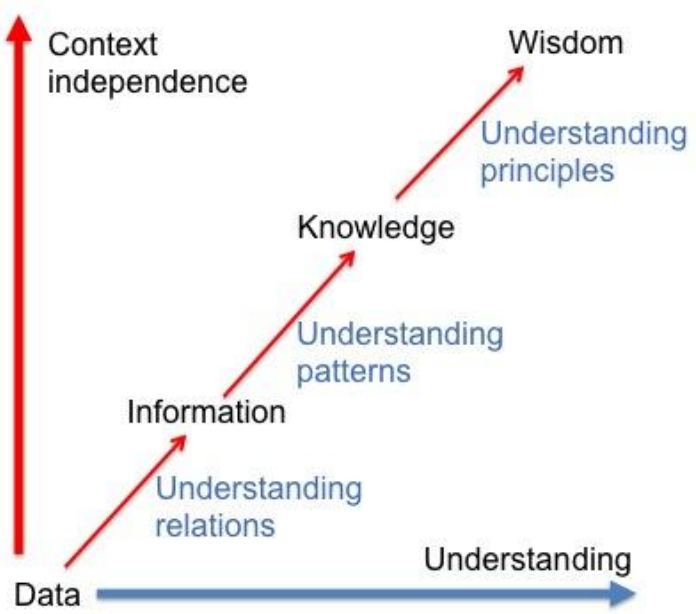


**Figure 3.2 information elaboraton with reference to the DIKW pyramid.**

The six dimensions for the representation of the gestural interaction in a smart environment are in the lowest layer of the pyramid. On the other layers, the data can be elaborated using different relations and principles. The choice of such parameters depends on the application. The spatial model for the characterization of the gestural interaction is important and a first measure that can be calculated from the dimensions is the distance between the entities. In fact, other works suggested to model the area around the user as spatial region where it is possible to manipulate objects (Surie et al., 2007; Surie et al., 2010). The objects that are in the reach of the user can be manipulated; if these objects are augmented ones, then they integrate computational capabilities, allowing the user to interact with them through contact gestures. For instance, a user can

interact with smart surfaces through touch gestures and with smart artifacts through tangible gestures. The research showed that another way of interacting with the environment, i.e., with the smart objects, it is also through distance gestures, which are performed in the air. The problem associated to this kind of distance interaction concerns the identification of the target of the gesture. One solution could be creating a gesture taxonomy that presents a set of gestures for each smart object. In this case, every command is associated to a different gesture. This approach makes the number of gestures to increase with the number of possible commands. Increasing the number of smart objects augments the number of gestures that a user has to learn, which represents a problem in terms of memorability and cognitive load. The opportune use of the contextual information allows the adoption of a second approach. In this case, the elaboration of the data of the six dimensions can provide adequate information to understand the user's intention and focus. In fact, other researchers already emphasized the issue and the importance related to identifying the user's focus of attention for natural interaction in smart environments (Shafer et al., 2001). This information can help to disambiguate the aim and the meaning of a specific gesture performed in a determined situation. For example, the orientation and the distance of the object can determine the target of the command (Figure 3.3).

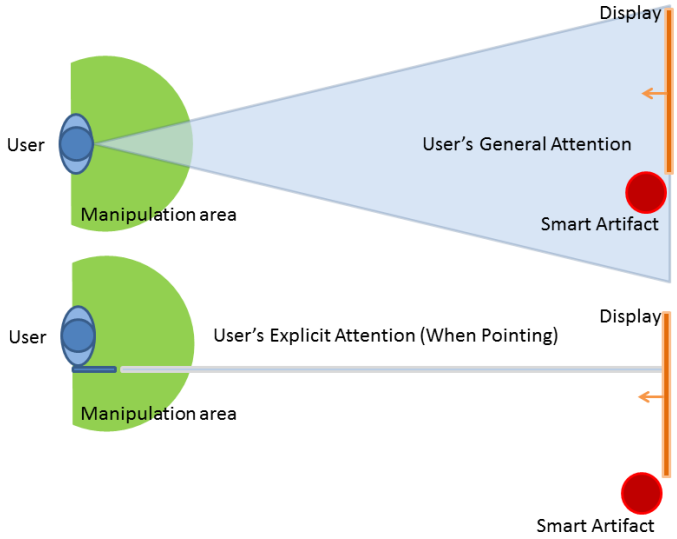


**Figure 3.3 Example of spatial model for the human-environment interaction.**

In Figure 3, the spatial model allows the representation of the user in the environment and the orientation and the distance allows understanding the focus of the general attention. The movement, in this case a deictic gesture, can help the system to identify the specific target of a command. The possibility of identifying the user's focus brought to the creation of a novel concept called functional gestures.

## 3.3 Functional Gestures

In (Bub et al., 2008), functional gestures have been defined as "gestures associated with the conventional uses of objects." In this thesis, this definition is extended to include generic smart objects. In fact, the model here presented classifies the interactive entities according to the following 2-elements taxonomy: two-states smart objects and complex smart objects.

Two-states smart objects are simple entities that are characterized from having just the states ON and OFF. Lamps could typically belong to this category.

Complex smart objects can be modeled by a more or less elaborate state machine representation in which each state defines the links between gestures and actions.

It is convenient to distinguish the two-state entity class for the wide availability of devices that can fit this class and the fact that two-state entities do not need pre-configuration. On the other hand, the state machine representation of the functionality of a complex entity is strictly linked to the functions of the device. Automatic state-machine generation, configuration and deployment are not the focus of this research but solutions based on an ontological description of the interactive entities, such as presented in (Sommaruga et al., 2011), can help this process: ontologies can abstract heterogeneous devices as homogeneous resources.

A function is an action triggered on an abstraction of an interactive entity. Examples of functions are start, next element, undo, etc. A functional gesture/command is strictly connected to the functionalities of the smart objects that the user is interacting with. For instance, the next element command has no meaning for an entity with just two states.

The proposed model aims to enhance the interaction between the human and the smart objects finding a good balance between cognitive load and vocabulary expressiveness, in the context of gesture-based interaction. Interaction in smart environments and gestural interaction can be very varied; in order to address these challenging issues and focus on this research, some constraints have been fixed.

**Interaction lexicon should:**

1. Have a moderate number of gestures, to reduce the cognitive load for the user that has to recall the interaction to perform. E.g., seven more or less two is the range of numbers suggested in (Miller, 1956) and it has been used in this scenario (eight).
2. Define a set of meanings and functions and not the cinematic and dynamic of the gesture itself that the user can freely choose. Such meaning should be generic. Based on the research presented in (Neßelrath et al., 2011), the gesture meanings are based on their functions: the functional gestures.

**Environmental feedback interfaces should:**

3. Be designed to be compatible with the generic meaning of the gesture vocabulary and increase the intuitiveness of the interaction.

Functional gestures are dynamically associated to precise actions on the entities present in the environment exploiting contextual information. In this thesis, two additional types of contextual information are used beyond the six dimensions of the spatial model: the system status and the user's activity. These concepts will be further investigated in Chapter 5 and Chapter 6.

In order to provide a better understanding of this approach, a specific scenario is presented. A user is in a smart living room and there are 4 different two-states smart objects. These objects are a work lamp, a floor lamp, a fan and a media center (Figure 3.4). The possible functions are for every object is "Turn On" or "Turn Off"; this means that the number of total actions is eight.

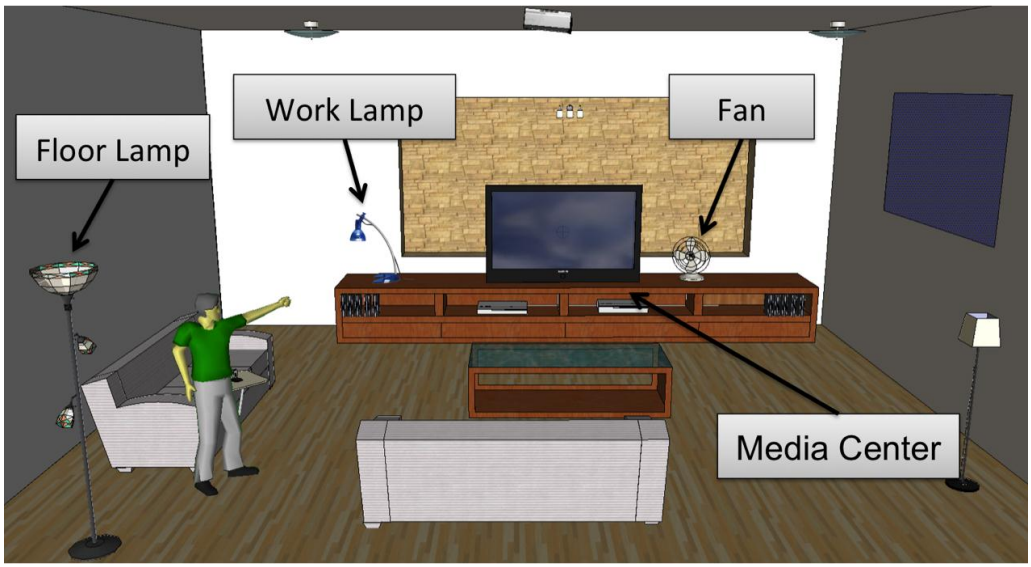


**Figure 3.4 The human-smart environment interaction scenario.**

Adopting the "simple approach" with a relation one-to-one between the gesture and the action would need eight different gestures for the eight commands: Turn On and Off the Media Center, Turn On and Off the Work Lamp, Turn On and Off the Floor Lamp, and Turn On and Off the Fan.

If the system is "entity-aware", it allows tracking the user and calculating the focus of attention understanding the aim of the command. In this case, the

gesture taxonomy would need two gestures: one for the action “Turn on” the target smart object, and one for “Turn Off” the target smart object. In fact, the system automatically recognizes the target of the focus attention through the contextual information and the smart object will automatically execute the command associated to the specific gesture.

The third approach, the one proposed in this thesis and called “functional”, allows using only one gesture. In fact, in this case the context-aware system takes into account also the state of the smart objects; hence, it is possible to find a general function that is “Switch State” with no regard to the final state. In this case, where all the interactive entities are two-state objects, it means that if the smart object is in the state “On”, the functional gesture will make it switch to the “Off”; vice versa, if the smart object is in the state “Off”, then the functional gesture “Switch” will make the smart object to go in “Off”. The smart object that has been identified by the system as the target of the focus of attention of the user will execute the command taking into account the current state. Table 3.1 reports the number of commands for every presented approach.

**Table 3.1 Number of commands for the three presented approaches.**

| Smart Object | Simple approach | Entity-aware | Functional |
|---|---|---|---|
| **Media Center** | 2 | 2 | 1 |
| **Work Lamp** | 2 | = | = |
| **Floor Lamp** | 2 | = | = |
| **Fan** | 2 | = | = |
| **Total** | 8 | 2 | 1 |

The functional gesture approach is based on an opportunistic context-aware model conceived to augment the expressivity of a small lexicon of gestures. The small size of the lexicon reduces the impact on the user cognitive load, whereas the functional gesture approach augments the vocabulary expressivity, with the results of increased expressivity and reduced cognitive load. Moreover, a

reduced number of gestures usually improves the accuracy of the gesture recognition systems based on classifiers using machine learning algorithms; in fact, a classifier that recognizes a small number of gestures generally outperforms the same system trained on more gestures, as reported in (Wachs et al., 2011).

The functional gesture approach does not provide a guideline to design the movement, but it is an approach for the development of gesture recognition systems that can improve the user experience through a reduced taxonomy. As already stated, a smaller vocabulary of gestures grants a lower cognitive load, a higher learnability and helps developers to have classifiers that performers better. This approach helps the designers and the developers to map the functions available in the environment to the available commands that can be given to the interactive entities. After that, the designers have to associate a gesture to each function. This approach does not provide a guide for the gesture design and the scientists can choose to follow the preferred method (e.g., as with the Wizard of Oz (Akers, 2006)), or role playing (Nielsen et al., 2004), or making the users choose their gestures (Oh & Findlater, 2013)). This model can be reinterpreted using the DIKW pyramid to show how the elaboration processes distribute on the different layers (see Figure 3.5).

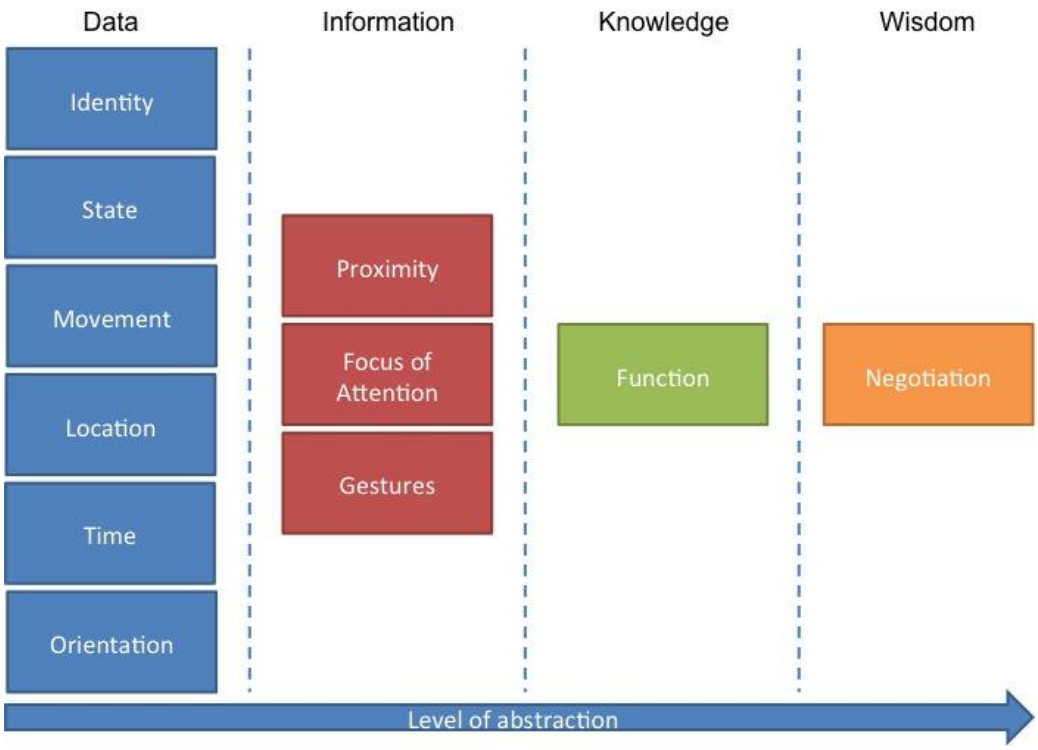


**Figure 3.5 The context information elaboration represented on the DIKW layers with reference to the gestural interaction scenario.**

The six dimensions that can be retrieved by the sensors are put in the Data layer. In the upper layer, the dimensions are elaborated in order to obtain important spatial information; the focus of attention, the proximity and the performed gestures are calculated in the case of the functional gesture approach. This information is further elaborated in order to identify the target of the interaction and to recognize the performed gesture; the successful elaboration of this information allows the system to execute the command meant by the user. The functional gesture is represented in the Knowledge layer, since this is the interpretation of the information to find a behavioral pattern in the space to model the interaction. The top layer is called Wisdom and in this model contains the negotiation between the user and the system. In fact, in this layer other parameters that do not concern directly the spatial representation of the interaction are involved. In this thesis, these parameters are the system status and the user's activity. The system that tracks this further information and creates a history of it, can proactively suggest commands or filtering information

or avoiding dangerous activities. The negotiation process can be modeled as a dialog, implementable as verbal or nonverbal conversational behaviors. In fact, Bickmore and Picard claim that to build and to maintain a long-term human-computer interaction require, at a minimum, some kind of natural conversational interface (Bickmore & Picard, 2005). Therefore, the negotiation between human being and smart environment can be the simple command interpretation and associated task execution or can become a conversation to reach an agreement. The negotiation dialogue is not addressed in the frame of this thesis.

An ecology where the objects and users are described taking into account the aforementioned six dimensions can be represented as in Figure 3.6.

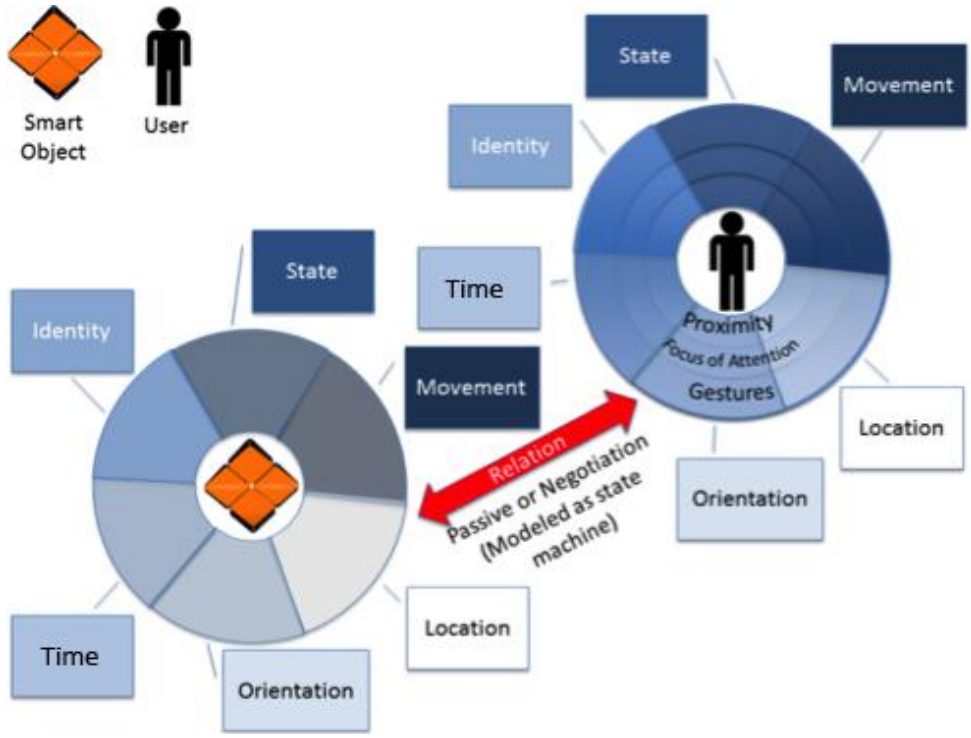


**Figure 3.6 The representation of the information based on the user and the smart object.**

The amount of data to treat in this ecology grows with the number of entities that are present in the environment. During this thesis, it was not possible to treat all the information included in the framework and the focus is on a simplified model that takes into account some selected dimensions. In particular,

the entities will include a status; the relation will be described through the spatial dimensions including orientation, location and time. The status of a user includes the information concerning the activity, which will be treated in Chapter 5; the status of a smart object can be described as on or off for the two-states objects. However, this concept will be pushed further expanding it to the entire system in Chapter 6. In this chapter, the system based on the synergistic type of the pervasive paradigm will be described using the concept of status, which allows optimizing the gesture recognition in order to improve a better user experience.

## 3.4 Summary

In this chapter, a novel framework for the development of gestural interfaces of the ubiquitous computing era has been presented. The main advantages provided by this high-level framework are two. The first advantage consists of the possibility of structuring the context information for gesture recognition in smart environments. In fact, in the subsection 2.3.3 of the literature review it has been reported Norman and Nielsen's work, where they pointed out that a more systematic approach to the design and development of gestural interfaces is still missing. Some works introduced new important concepts as the spheres of roles, the proxemics model and Karam's classification (Karam, 2006); these concepts actually provided meaningful insights about the development of gestural interfaces but without leaving a specific guideline to follow. The framework presented in this thesis specifically addresses this problem of a lack of systematic guidelines for the design and development of gestural interfaces presenting a high-level framework that aims at facilitating structured organization and use of the contextual information. In particular, this framework has been conceived to take into account all the possible types of gestural interaction with the environment, i.e., touch gestures, tangible gestures and distance gestures.

The second advantage is that this framework is suitable for the implementation of the functional gestures, which allow optimizing gesture taxonomies. The

functional gesture concept allows using a novel method that is based on the aforementioned framework to map the gestures to the functions. The introduction of this method fills a gap between the gesture classification and the gesture design presented in the subsection 2.3.3 of the literature review. In fact, the aim of this method is to allow using the contextual information in order to create a system with a reduced number of gestures but that covers all the available functions. This allows developing and designing gestural interfaces with a reduced cognitive load and an augmented learnability impacting directly on the usability of the overall system.

# Chapter 4: Gesture Recognition Algorithms

## 4.1 Introduction

The previous chapter introduced a novel framework for the context-aware gestural interaction in smart environments. Following the user-centered approach, it is important to iteratively develop a proof-of-concept to measure the usability of a gestural interface implemented following the guidelines provided by this framework and using the method based on the functional gesture concept. The literature review highlighted that it is important to detect and recognize gestures unobtrusively and without limiting the user's freedom of movement in the environment. The unobtrusive approach imposes the adoption of vision-based technologies, which introduces a new challenge: developing specific algorithms for the view-invariant gesture recognition. In this chapter, a novel technique incorporating contemporary methods and technologies for the view-invariant recognition of dynamic and deictic gestures is presented. This novel technique integrates a procedure for the calibration of depth cameras, an algorithm for the deictic gesture recognition and the implementation of machine learning techniques for the dynamic gesture recognition.

## 4.2 Developing the proof-of-concept

Youngblood et al. defined a smart environment as one that is able to acquire and apply knowledge about the environment and its inhabitants in order to improve their experience in that environment (Youngblood et al., 2005). Designing a

smart room that can achieve this goal involves the context awareness and the possibility of interaction with the people. In particular, gestures are a natural way of interaction for humans and integrating these commands with information coming from the situation can make the environment to support user's tasks. A smart room that can achieve this goal has to recognize the inhabitants' activity, it has to understand the direct commands ordained by the users and it has to integrate many smart objects to communicate with. The framework presented in this thesis allows developing a system that can provide a context-aware gestural interaction that not only includes the modeling of both contact and distance gestures but it also provides a procedure for the gesture taxonomy design. This procedure allows mapping the gestures in functions reducing the number of gestures; therefore, it reduces the cognitive load and increases the memorability.

The context information comes from the data related to the states and positions of the smart objects, and to the tracked inhabitants' postures and movements. This information is modeled in a 3D virtual space and the spatial relation allows recognizing the gesture and the related function. The literature showed that the adoption of vision-based technologies facilitate sensing the spatial information in a smart environment and recognizing gestures (Wachs et al., 2011). In particular, vision computing allows detecting movements in an unobtrusive manner, without the need of putting any device on the user. In HCI design, this is a principle called "come as you are", which that poses no requirement on the user to wear markers, gloves, or long sleeves, fix the back- ground, or choose a particular illumination.

### 4.2.1 Microsoft Kinect

Various techniques are used to acquire 3D information using video-based approaches. It is possible to use a single RGB camera or multiple calibrated RGB cameras; in this case the problem usually refers to the lighting condition changes. Other types of cameras that acquire directly 3D information exist: the laser scanners, the Time-of-Flight (ToF) cameras and the structured light

cameras. They are based on different measurement principles and they present different advantages. The laser scanners are rarely used in real-time gesture recognition because of the long acquisition time. The ToF cameras and the structured light cameras have similar technical characteristics but very different prices. In fact, their measurement principles are different: the ToF camera illuminates the scene with short infra-red light pulses and the system measures the time taken until the reflected light reaches the camera again; as the measured times are directly proportional to the distance, the camera gives a distance value for each pixel. The structured light camera uses an infrared laser emitter that projects a specific pattern on the scene and an infrared camera, which allows measuring the depth information through a triangulation process. In particular, the Kinect launched by Microsoft in the late 2010 brought a novel interest in the gesture-based interactions in both the academy and the industry. The main reason for its success is due to the very low price that made the 3D cameras very affordable and, therefore, positioned to become ubiquitous.

The Microsoft Kinect is composed of an infrared laser projector, an infrared camera for the depth information sensing and an RGB camera (Figure 4.1). The Kinect provides 30 depth images and a 30 RGB images per second. The depth information is represented with 11 bits for 2,048 levels of sensitivity.


**Figure 4.1 The Microsoft Kinect device integrates an RGB camera and a depth camera.**

The Kinect depth sensor range is minimum 800 mm and maximum 4000 mm with 43° vertical by 57° horizontal field of view. The Kinect for Windows Hardware can however be switched to Near Mode which provides a range of 500 mm to 3000 mm instead of the Default range. Technical experts evaluated the Kinect accuracy and estimated that the random error of depth measurements increases quadratically with increasing distance from the sensor and reaches 4 cm at the maximum range of 5 meters (Khoshelham & Elberink, 2012). The depth resolution also decreases quadratically with increasing distance from the sensor (see Figure 4.2, which has been extracted from (Khoshelham & Elberink, 2012)). The point spacing in the depth direction (along the optical axis of the sensor) is as large as 7 cm at the maximum range of 5 meters (see Figure 4.2).

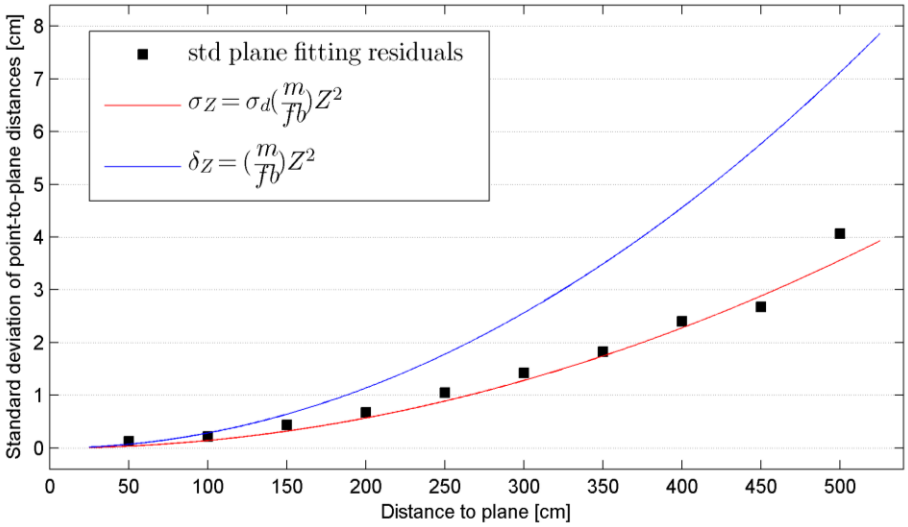


**Figure 4.2 Standard deviation of plane fitting residuals at different distances of the plane to the sensor. The curves show the theoretical random error (red) and depth resolution (blue); (Khoshelham & Elberink, 2012).**

Using the infrared camera, Kinect can recognize up to six users in the field of view of the sensor. Of these, up to two users can be tracked in detail. An application can locate the joints of the tracked users in space and track their movements over time. Skeletal Tracking is optimized to recognize users standing or sitting, and facing the Kinect; sideways poses provide some challenges regarding the part of the user that is not visible to the sensor. Using multiple

Kinects illuminating the same area should avoid the occlusion problem making possible to recognize the gestures performed in sideways poses.

### 4.2.2 Using Multiple Kinects

Using multiple Kinects to illuminate the same area involves creating interference between the infrared laser patterns that are at the base of the functioning of this device. Each Kinect projects its own infrared pattern for the calculation of the depth information and interferences can degrade the information quality creating black spots on the 3D image. In Figure 4.3 it is possible to see the infrared dots of the Kinect pattern; on the left only one infrared projector is activated, on the right two infrared projectors are active: it is possible to notice the augmented number of dots, which creates the interference.

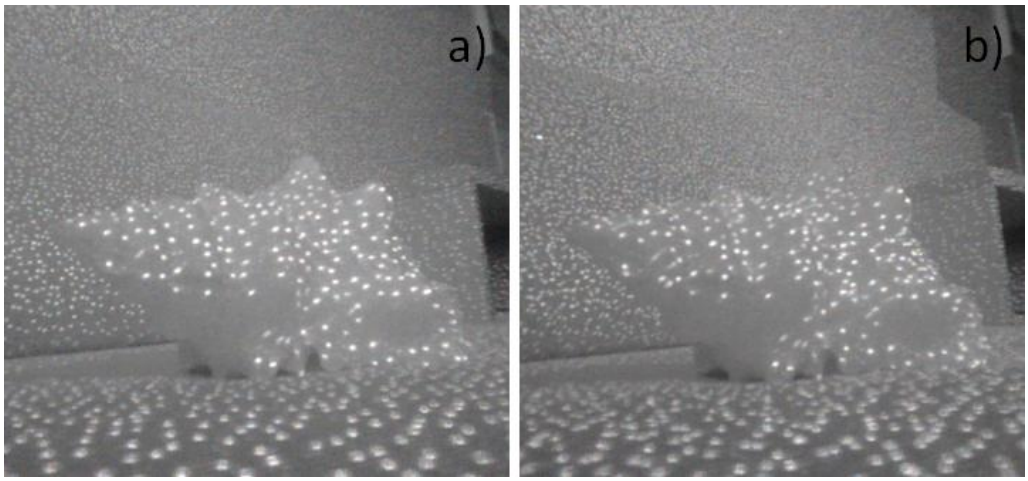


**Figure 4.3 a) The laser pattern projected by 1 Kinect, b) Interference between the infrared laser patterns of 2 Kinects: the number of dots is increased.**

In order to assess if the interferences change significantly referring to the number of active Kinects and their positions, 5 different configurations have been tested. Figure 4.4 represents the camera configurations that have been tested. The Kinects have been positioned in A, B and C. The colored triangles represent the field of view of the cameras from the A, B and C positions. The striped areas of the triangles represent the interactive areas, or rather, the areas

where people can be easily tracked. This suggested area begins at a distance of 1 m from the Kinect and arrives to 3.5 m. A person was present in the test scenario and he was positioned on the white circle in the center of the figure. The A, B, and C positions are at the same distance from the person, in order that the points of the patterns projected from the infrared lasers have same brightness and dimensions on the person. The optical axis of the Kinect positioned in A intersects the optical axis of the Kinect positioned in B forming an angle of 45°. The optical axis of the Kinect positioned in A intersects perpendicularly the optical axis of the Kinect positioned in C. In configuration 1 there was only one active Kinect and it was positioned in A. In configuration 2 there were two active Kinects and they were both positioned in A. In configuration 3 there were two active Kinects, one was positioned in A and the other one in B. In configuration 4 there were two active Kinects, one was positioned in A and the other one in C. In configuration 5 there were three active Kinects, one was in A, one in B and one in C.

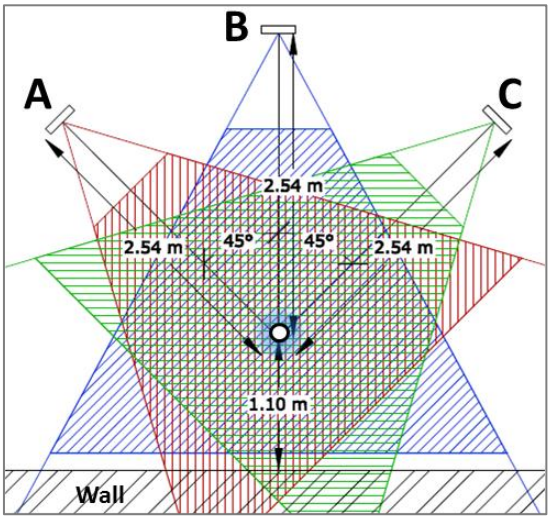


**Figure 4.4 Representation of the interference test configurations.**

To quantify the interference effect, the number of pixel without depth information has been calculated. Since the pixels without depth information change during time also on a static scene, then this number has been calculated making an average on 1000 frames for every configuration. The depth sensor of the Kinect captures 640x480 pixel frames; therefore, every frame has got 307200

pixels with depth information. For the configuration 1, an average of 5325 pixels without depth information has been calculated (with standard deviation of 165 pixels); for the configuration 2 the average was of 14018 pixels and the standard deviation was of 404 pixels; for the configuration 3 the average was of 12502 pixels and the standard deviation was 319 pixels; for the configuration 4 the average was of 13000 pixels and the standard deviation was of 295 pixels; for configuration 5 the average was of 21813 pixels and the standard deviation was of 432 pixels. The graph depicted in Figure 4.5 reports the aforementioned values. After these tests, it has been verified that the interference caused by two Kinects is not significant for the skeleton tracking and it remains almost constant regardless the relative position of the two cameras. However, using two Kinects in configuration 4 permits capturing the tracked users' movements from very different perspectives. This configuration permits to capture a very big portion of the users' bodies avoiding in many cases the occlusion of some limbs.

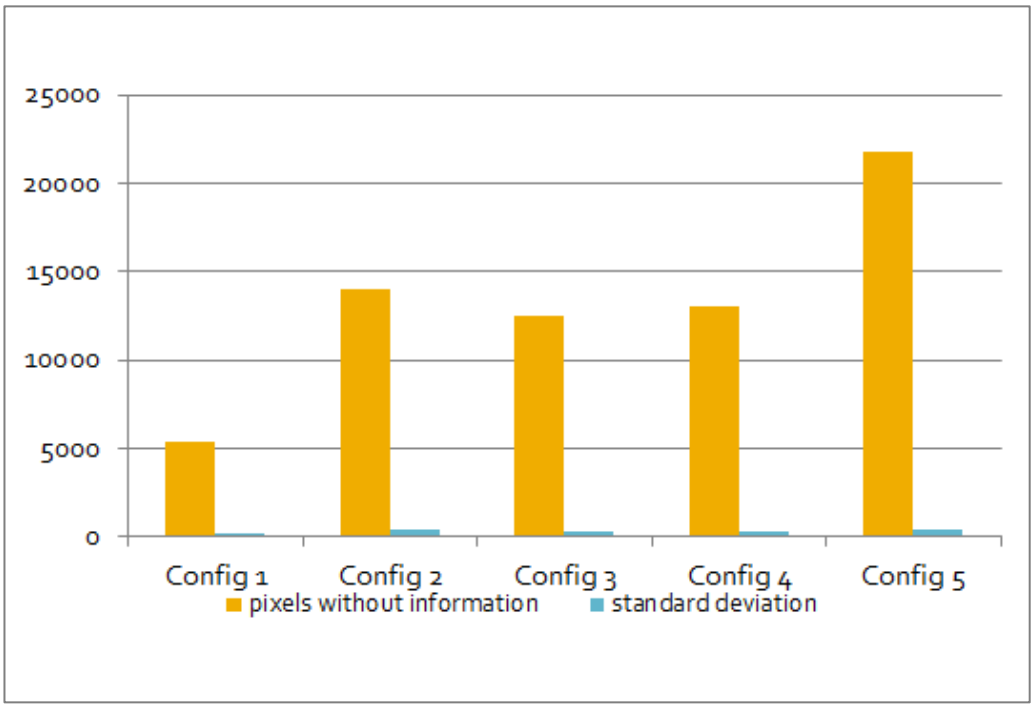


**Figure 4.5 This graph shows the number of pixel without information in the five different configurations considered during the tests.**

Using three Kinects in configuration 5 doubles the number of the pixels without depth information generated by the interference but it does not add significant

information for the user's skeleton re-construction. Figure 4.6 reports some pictures taken during the tests to provide a qualitative visualization of the interference in the different configurations.

According to the results of these tests, it has been decided to use two Kinect cameras positioned as in configuration 4.

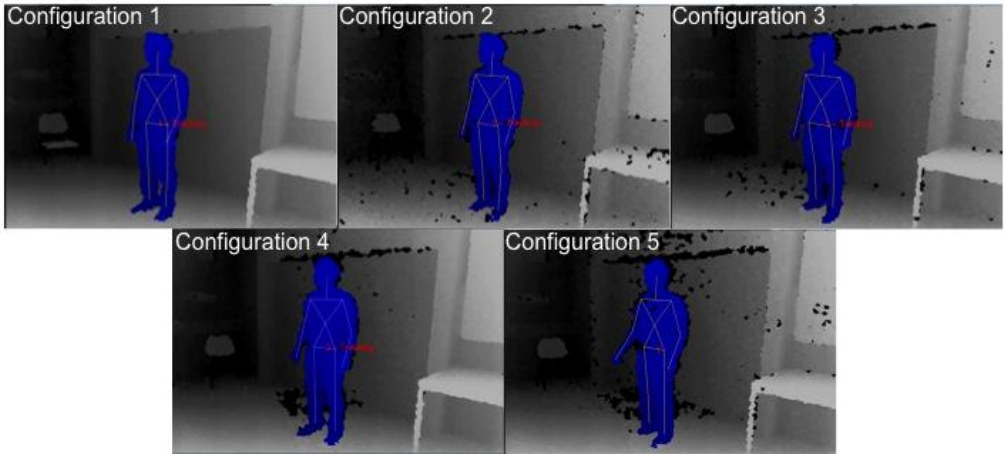


**Figure 4.6 Scene captures during the interference test for every configuration.**

The cameras calibration is crucial to reconstruct a 3D model using simultaneously multiple depth cameras. The calibration procedure is composed of two consecutive steps. The first one consists of the calibration between the IR and the RGB cameras in each Kinect. The second step is the calibration between the two different Kinects.

#### 4.2.2.1 Calibration between IR and RGB cameras

The acquisition of the common points in the 3D space by the Kinects is obtained using a simple checkerboard as shown in Figure 4.7.

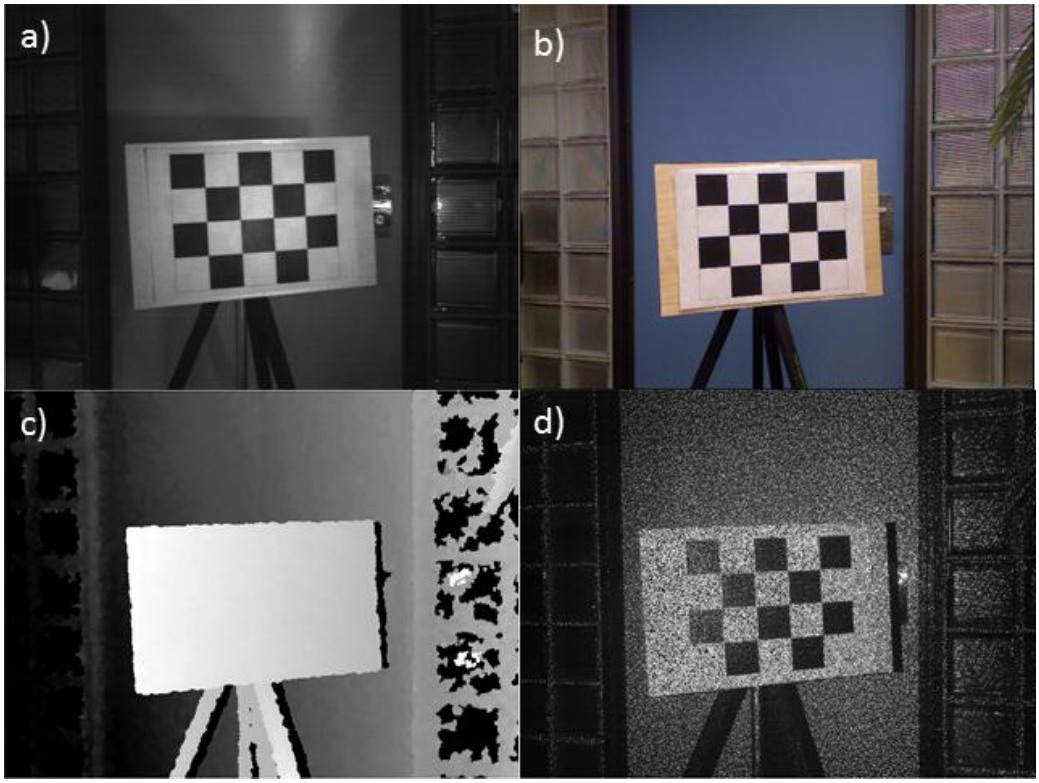


**Figure 4.7 Calibration between the IR and RGB cameras of the Kinect; a) the IR image with an IR source; b) RGB image of the same scene; c).depth image of the same scene; d) IR image with the unblocked IR laser projector.**

When the checkerboard is seen by both the Kinects, its center is calculated processing the RGB cameras images with OpenCV. Afterwards, the extracted points for every synchronized image are associated to the depth value extracted by the depth cameras. In order to obtain the depth value of each pixel captured by the RGB cameras, it is necessary to execute the calibration of the RGB camera and IR camera for every Kinect. For this procedure, it has been extended the work presented in (Van den Bergh & Van Gool, 2011) for the calibration of a RGB camera and a ToF camera. In order to obtain good corner detection the IR laser projector of the Kinect must be blocked and the checkerboard must be suitably illuminated by an IR light source (e.g., a halogen lamp), Figure 4.7 a). The intrinsic and extrinsic parameters of the RGB and IR cameras have been calculated using the Matlab camera calibration toolbox. Indeed, the pixel pRGB (expressed as a 2

× 1 matrix) has been calculated in the RGB image coordinates as (given that Z' is not zero).

$$p_{RGB} = \begin{bmatrix} x_{RGB} \\ y_{RGB} \end{bmatrix} = \begin{bmatrix} f_{x,RGB} \cdot {X'}/{Z'} + c_{x,RGB} \\ f_{y,RGB} \cdot {Y'}/{Z'} + c_{y,RGB} \end{bmatrix}$$

Where X', Y' and Z' are the 3D coordinates with respect to the RGB camera.

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = z_{IR} \cdot R \cdot \begin{bmatrix} f_{x,IR}^{-1} & 0 & c_{x,IR} \cdot f_{x,IR}^{-1} \\ 0 & f_{y,IR}^{-1} & c_{y,IR} \cdot f_{y,IR}^{-1} \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_{IR} \\ y_{IR} \\ 1 \end{bmatrix} + T$$

The pixel pIR is represented with the xIR and yIR coordinates in the IR image and with the associated depth value (zIR). R is the rotation matrix and T is the translation matrix. The f and c coefficients rep-resent the intrinsic parameters. Finally, the depth value corresponding to the location pRGB in RGB image is Z'.

#### 4.2.2.2 Calibration between two Kinects

Every Kinect camera measures the distance to each visible point on an object to create a collection of distances called depth map. The 3D points captured by a single Kinect are expressed referring to its own reference frame that has the origin in the Kinect's depth camera with the z-axis pointing out of the camera (Figure 4.8).

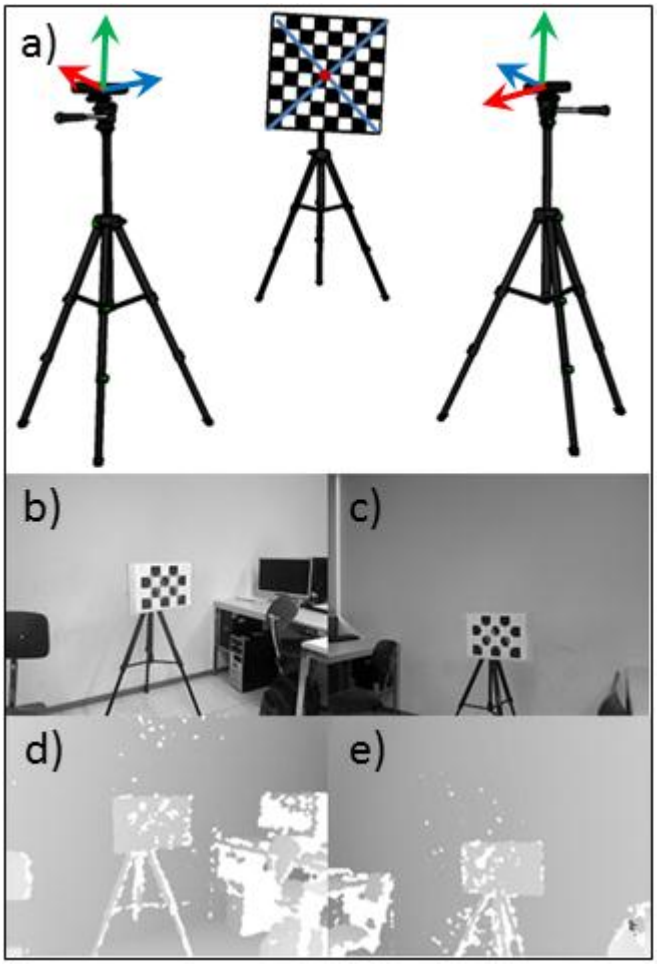


**Figure 4.8 Calibration between two Kinects using a checkerboard; a) disposition of the Kinects and the checkerboard; b) the RGB image captured by the Kinect on the left and elaborated with OpenCV; c) the RGB image captured by the Kinect on the right and elaborated with OpenCV; d) depth image captured by the Kinect on the left; e) depth image captured by the Kinect on the right.**

For the calibration between two Kinects, it is necessary to acquire two 3D point sets that represent the same point in the 3D space viewed by the two Kinects. Introducing the formulas, pi is the first set of points with coordinates in the reference frame associated to the first Kinect; pi' is instead the set of the same points in the 3D space captured by the second Kinect. Here, pi and pi' are considered as 3 × 1 column matrices. Therefore, it is possible to represent the transformation of the points expressed in the coordinates of the first Kinect's reference frame to the second Kinect's reference frame as

$$p_i' = R \cdot p_i + T + N_i$$

Where R is a 3 × 3 rotation matrix, T is a translation vector (3 × 1 column matrix), and Ni a noise vector. The aim is to find the R and T matrices that minimize

$$E^2 = \sum_{i=1}^{N} \|p_i' - (R \cdot p_i + T)\|^2$$

It has been chosen to use a non-iterative algorithm, which involves the Singular Value Decomposition (SVD) of R. In (Arun et al., 1987), it has been demonstrated being more efficient in terms of time requirements for the computation of the number of points of interest. Following this approach, the centroids of the 3D point sets were calculated

$$p' \triangleq \frac{1}{N}\sum_{i=1}^{N} p_i'$$

$$p \triangleq \frac{1}{N}\sum_{i=1}^{N} p_i$$

And

$$q_i \triangleq p_i - p$$

$$q_i' \triangleq p_i' - p'$$

Then it is necessary to calculate the 3 × 3 matrix and to find its SVD

$$H \triangleq \sum_{i=1}^{N} q_i \cdot q_i'^T = U \cdot \Lambda \cdot V^T$$

Finally, it is possible to calculate

$$R = V \cdot U^T$$

And

$$T = p' - R \cdot p$$

Finally it is possible to calculate the R and T matrices that minimize E2 and they are used to calculate the coordinates transformation.

### 4.2.3 Gesture Recognition

The calibrated Kinect cameras allow collecting and modeling the 3D spatial information of the environment and of the tracked inhabitant. The 3D model of the environment includes the users' skeletons and the smart objects. The system recognizes the users' postures and pointing gestures from the coordinates of the joints in real-time. This recognition process is based on two different algorithms that elaborate the values of the joints and the relative distances between them.
Researchers often represent human body as an abstracted skeleton (called also stickman) composed of connected joints (e.g., Badler & Smoliar, 1979; Thalmann et al., 1996; Herda et al., 2000; Fossati et al., 2010; Ramey et al., 2011) and many others). Extracting joints coordinates from an acquisition system can model the human movements in a very effective way. As a standard does not exist, then the number and the name of the represented joints can vary depending on the reference system. In this system the user's skeleton model has 15 joints that are: head, neck, right and left shoulders, right and left elbows, right and left hands, torso, right and left hips, right and left knees, right and left feet. In this model the hands correspond to the wrists and the feet to ankles. The entire model is represented in Figure 4.9. PrimeSense has introduced this model with the library OpenNI for the Microsoft Kinect device.

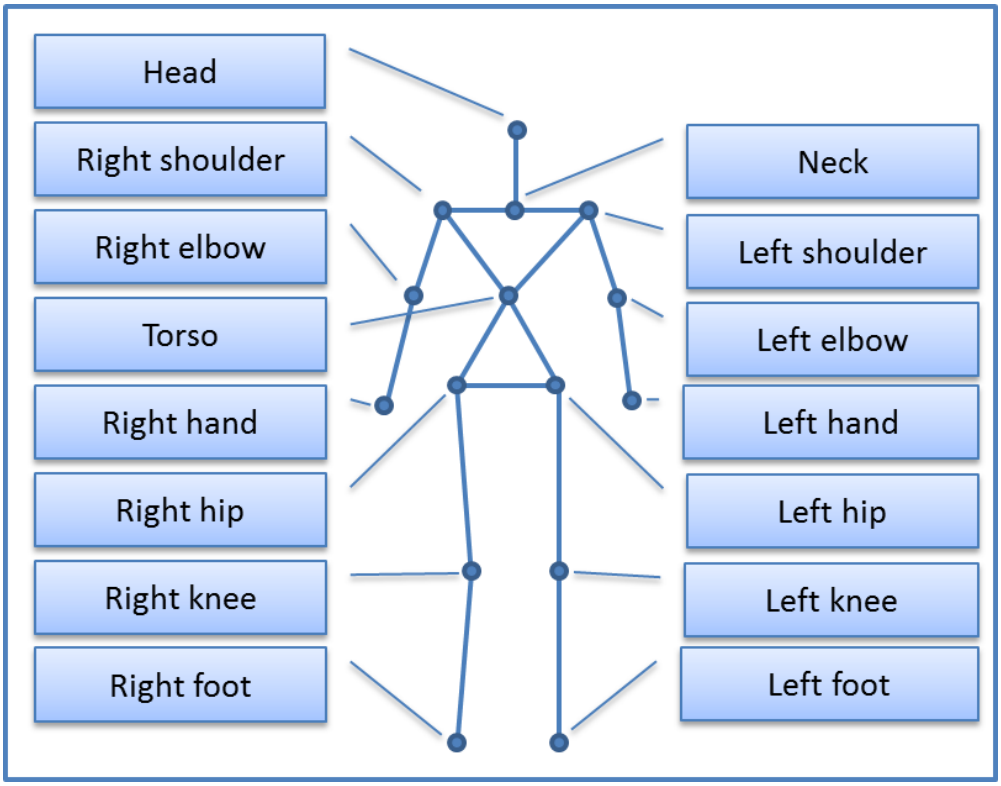


**Figure 4.9 Skeleton model with the labels of the tracked joints.**

The OpenNI libraries allows collecting the user's joints data coming from the two Kinects; then, it is possible to apply the transformation matrix to represent both the 3D skeletons in the same spatial reference frame. Once the two 3D skeletons are calculated, it is executed the fusion of the data concerning the tracked user to create a unique 3D skeleton. The joints that are considered during the fusion process must have the maximum value of the associated reliability factor that is provided by the OpenNI libraries. The result of this fusion process is a complete 3D user model composed of the merged user's skeleton coordinates and it allows overcoming problems due to the relative cameras and user positioning. In particular, it allows reducing the self-occlusion problem, which is the impossibility of the cameras to track some of the user's skeleton joint because some body parts are not visible from the camera since they are hidden by other body parts. Another problem can be related to the relative field of view of the cameras. In fact, when the user is too close to a camera, it will not be possible to track all the user's skeleton joints because some body parts are not visible. Using

two opportunely placed cameras, when the user is too close to one camera, the other camera can keep tracking the user joints and the fusion can provide a unique complete skeleton. In Figure 4.10 a), it is possible to see that the calibrated Kinects are tacking the same user and in one field of view the legs are not visible but in the other are. In Figure 4.10 b), it is possible to notice that the red skeleton misses the 3D coordinates of the joints corresponding to the user's legs; on the other hand, the green skeleton provides the 3D coordinates of the entire skeleton. The fusion process allows compensating this lack of information of the red skeleton using the coordinates coming from the camera that is actually able to track all the joints. The result of the fusion process is depicted in Figure 4.10 c), where the white skeleton represents the complete skeleton that will be used for the gesture recognition.

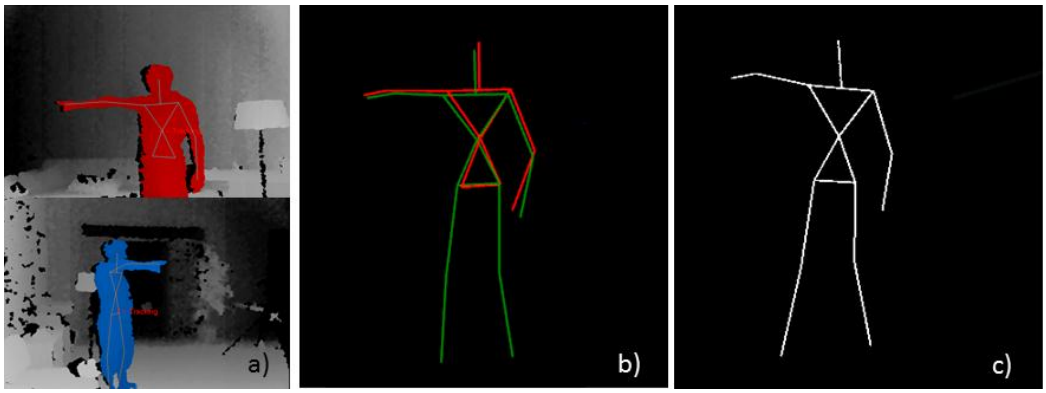


**Figure 4.10 Modeling the user information: a) user's skeleton in the two Kinect views; b) 3D model of the user's skeletons captured by the two Kinects; c) 3D fusion of the user's two skeletons in one skeleton.**

The two algorithms for the gesture recognition treat the data in order to allow view-invariant gesture recognition. That means that the gesture recognition process is independent from the camera viewpoints. In fact, the scenario of reference involves a smart living room that allows the user to interact with several devices distributed in the room (e.g., lamps, TV, hi-fi, et cetera) through gestures. The user can vary her/his position and orientation, but the system should grant continuous 3D gesture recognition for a seamless interaction

experience. The first algorithm is for the deictic gesture recognition and the second algorithm is dedicated to the dynamic gesture recognition.

#### 4.2.3.1 Deictic Gesture Recognition

The pointing gesture recognition is related to the right arm. The system needs to see 4 joints to interpret the pointing gesture: right hip, right shoulder, right elbow and right hand (Figure 4.11). On these four joints, a formal model with four angles is calculated to recognize the arm posture. Determining some angles constraints, it is possible to recognize the pointing posture.

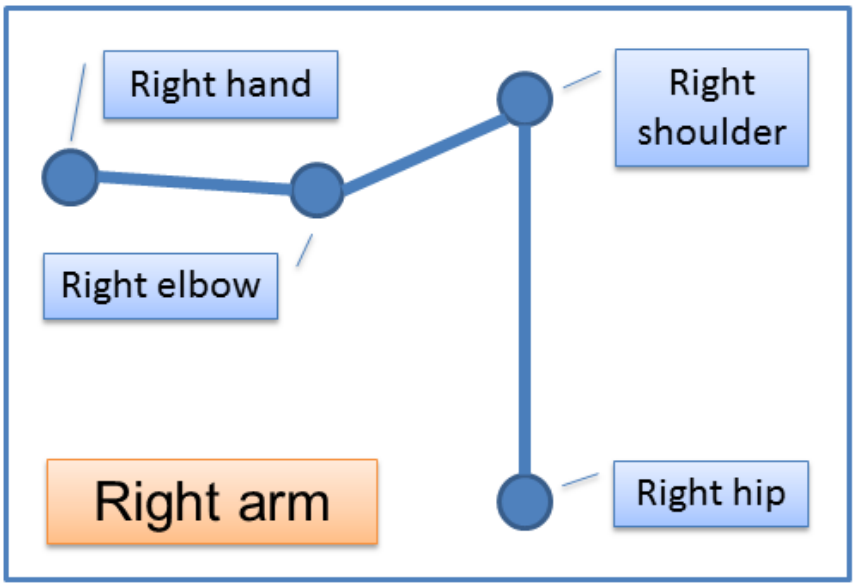


**Figure 4.11 Reference joints of the right arm for the pointing gesture recognition.**

The right shoulder, elbow and hand joints are represented in a Cartesian 3D coordinate system. X is in the lateral direction, Y is forward, and Z is up. The origin of this coordinate system is at the shoulder. Four angles are required to define the posture of the arm in this coordinate system: three resulting from rotations at the shoulder joint and one at the elbow. The angles used to define the arm rotation are illustrated in Figure 4.12 (Soechting & Ross, 1984; Soechting et al., 1986; Soechting et al., 1995). The arm posture is defined as the result of three successive rotations, starting with the upper arm vertical (along the Z-axis) and the arm in the parasagittal (Y-Z) plane passing through the shoulder (if the

forearm is not fully extended). The first rotation (η) is about the vertical Z-axis and determines the yaw angle of the arm. The second rotation (θ) is about an axis perpendicular to the plane of the arm (the lateral, X-axis if there is zero yaw) and determines the arm's elevation. The third rotation (ζ) is about the humeral axis. This rotation does not change the location of the elbow but does affect the location of the hand in space and the plane of the arm. The ϕ angle is defined as the angle of flexion of the forearm, ϕ = 0 corresponding to full extension.

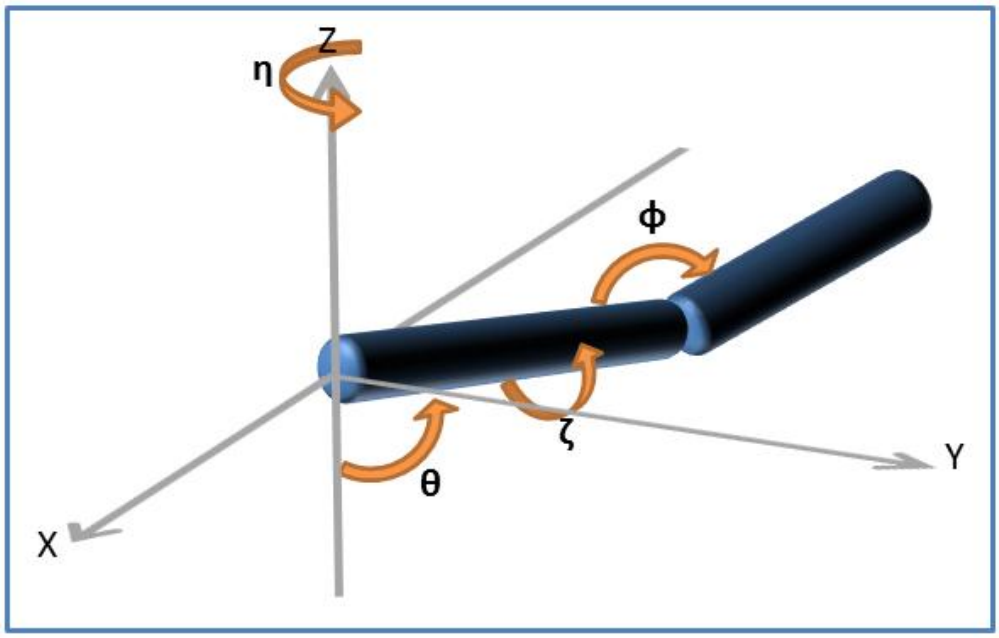


**Figure 4.12 Arm posture modelling.**

To recognize the arm posture 3 simple constraints have been set using a simple decision tree. The elevation angle must be between 65 and 95 degrees (65<θ<95) and then the angle of flexion of the forearm must be bigger than 135 degrees (ϕ>135). Figure 4.13 shows the decision tree of the deictic gesture recognition algorithm.

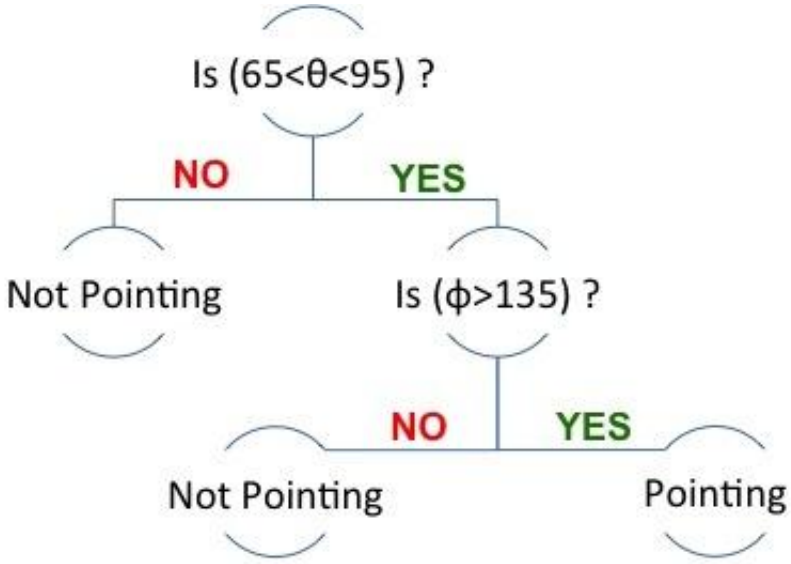


**Figure 4.13 Decision tree for the pointing gesture recognition**

There is not a maximum angle of flexion of the forearm because of the natural limit of this movement; in fact this angle cannot go beyond 180 degrees. Because of the many degrees of freedom of the human arm, this configuration of angle constraints allows the user to point in many directions (remaining with the arm near to the XY plane) and to keep the arm extended (also if not fully extended to make the gesture less tiring) with no regard to angle ζ; so the user can reach this movement from a starting arm posture in a natural way. In fact, the current arm postures depend from the starting posture violating the Donders' law (Soechting et al., 1995).

When the decision tree is in the "Pointing" state, the system elaborates the data in order to recognize the target of the gesture. The virtual 3D model constructed on the information gathered by the Microsoft Kinects allows identifying the selected smart object. The information level of the context allows reasoning with a spatial model and the gesture recognition is based on the popular "ray casting" technique (Dang, 2007). This selection technique is based on the mechanism known as "laser gun" selection that was introduced in (Liang & Green, 1994). A light ray is emitted from the user's hand. The user can control the starting point and the orientation of the ray. This technique allows the user to select objects

beyond the area of normal reach. The user points at a target through a virtual ray of light that is the extension of the user's dominant arm in the model. When the virtual light ray intersects the smart object, then the latter can be selected. This model uses radial base functions with the Mahalanobis distance to represent the smart objects as ellipsoids in the model. If x is the multivariate random variable, mu is the mean and S the covariance matrix, the Mahalanobis distance can be calculated as:

$$D_M(x) = \sqrt{(x-y)^T S^{-1} (x-\mu)}$$

The ellipsoids representing the smart object are projected on the 2D plane long the direction of the pointing ray. Since the pointing ray is projected onto a point, the best matching object can be determined by evaluating the projections of the base functions at that point. In order to simplify the calculation, all the smart objects are modeled as spheres and the covariance matrix can be modeled as the identity matrix; hence, the Mahalanobis distance is reduced to the Euclidean distance of the projection from the center. Otherwise, knowing the 3D coordinates of the two points of the user's arm represented in the 3D skeleton, called $x_1=(x_1, y_1, z_1)$ and $x_2=(x_2, y_2, z_2)$, it is possible to calculate directly the distance of the ray from the sphere center $x_0=(x_0, y_0, z_0)$ as following:

$$D = \frac{|(x_0 - x_1) \times (x_0 - x_2)|}{|x_2 - x_1|}$$

Where X denotes the cross product.

The smallest distance defines the selected object in the model. Figure 4.14 provides a visual representation of the spatial model with the line (the prolongation of the user's arm) intersecting one of the ellipsoids (the smart objects).

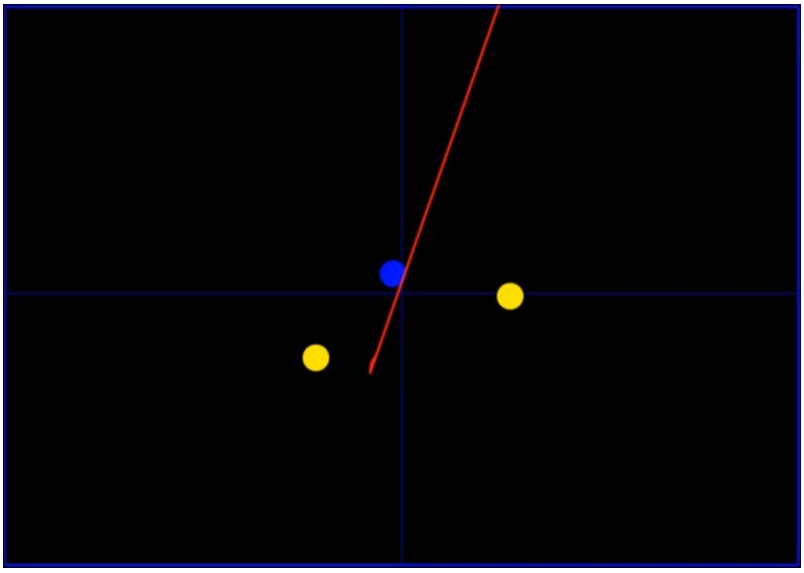

**Figure 4.14 Graphical representation of the spatial information: the circles are the smart objects, the thick red line is the user's arm and the red line is the ray for the calculation of the selected object.**

#### 4.2.3.2 Dynamic Gesture Recognition

There are two types of dynamic hand gestures: directional gestures and non-directional ones. Directional gestures are simple linear movements without changes of directions, while non-directional gestures are nonlinear such as circle and triangle (Yuan et al., 2010). This approach allows modeling and recognizing in real time both the types of dynamic gestures.

As explained in the previous sections, the calibrated Kinects allows tracking the user's joints. For the dynamic gesture recognition, the hands' movements are recorded over time as sequences of 3D coordinates that compose a 3D trajectory. In order to model these gestures in a view-invariant manner, the recorded 3D gesture trajectories are transformed from the spatial reference frame associated to the Kinects to the reference frame that has origin between the shoulders of the user's 3D skeleton.

A common approach for the analysis of gesture trajectories is the hidden Markov models (HMMs) (Rabiner, 1989). HMMs provide good representation properties for time series data and reach excellent results in various applications. During

the development of this proof-of-concept, the classifier has been developed using the Accord.NET libraries.

An advantage in using the HMMs is that it is necessary only to specify the number of states that are needed to model the input sequences and the model should be able to figure a suitable sequence of states to support the observations used as training. The HMMs work on the assumption that in any sequence in input, the current observation will only be dependent on the most immediate previous one; this principle is referred to as Markov probability. Given a sequence of observations $x = \langle x_1, x_2, \ldots, x_t \rangle$, in this case the 3D coordinates of the performed gestures, and a corresponding sequence of states $y = \langle y_1, y_2, \ldots, y_t \rangle$, the probability of any sequence of observations occurring when modeling on a given sequence of states can be stated as:

$$p(x, y) = \prod_{t=1}^{T} p(y_t \vdots y_{t-1})\, p(x_t \vdots y_t)$$

Where the probabilities $p(y_t \vdots y_{t-1})$ represent the probability of being currently in state $y_t$ and in state $y_{t-1}$ in the previous instant *t-1*. The probability $p(x_t \vdots y_t)$ is the probability of observing $x_t$ at instant $t$ given that the model currently is in the state $y_t$. To compute these probabilities, it is possible to use the two matrices A and B, which are stated as:

$$A = p(y_t \vdots y_{t-1})$$

$$B = p(x_t \vdots y_t)$$

The matrix A represents the probability of passing from one state to another; the Emission matrix B is the matrix of observation probabilities, which provides the distribution density $p(x_t \vdots y_t)$ associated to a given state $y_t$. The overall model definition can be written as the following tuple:

$$\lambda = (n, A, B, \pi)$$

Where $n$ is an integer representing the total number of states of the HMM; the matrices A and B have been already defined; the $\pi$ is a vector of initial state probabilities determining the probability of starting in each of the possible states in the model.

The algorithm for the calculation of the transition probabilities that was implemented in the Accord.NET libraries is the Baum-Welch algorithm (Welch, 2003). This algorithm allows training the HMMs in an unsupervised manner. The state-transition topologies implemented in this library are the ergodic and the forward-only. The ergodic topology represents an HMM where all states can be reached from any state; in the forward-only topology, the transition from the states can only go forward. Based on an empiric approach, it was identified the ergodic topology as the best choice since during the tests it provided the highest recognition accuracy. Probably, this topology allows a better modeling of dynamic non-directional gestures. In fact, in this case, the 3D trajectory performed by the hand movement is represented as 3D coordinates and the temporal dimension is not explicit (it can be retrieved as the differential sum of the different points). Therefore, the hand movement in non-directional gestures can go back to the previous values.

For the sequence classification, it was chosen to use the maximum likelihood decision rule. This means selecting the likelihood of the sequence as if it was the probability of the class given the sequence. The maximum likelihood decision rule can be stated as:

$$\hat{y} = \underset{\omega_j \in \Omega}{\operatorname{argmax}} P(x \vdots \omega_i)$$

Where the $\omega_j$ is associated to the estimated label of the sequence with the maximum probability. Each model $\omega_j$ provides an estimate of the probability $P(x \vdots j)$; therefore, it is possible to replace each $P(x \vdots j)$ with the likelihood given as output by the model.

Once implemented the HMM classifier, it is sufficient to provide a training set of gestures (i.e., the transformed 3D trajectories), which usually consist of some repetition per type of gesture in order to allow the training of the hidden parameters. Once trained the HMM classifier, the 3D gesture trajectories fed to the classifier will be elaborated and it will be given as output the corresponding gesture type with the highest probability.

The diagram in Figure 4.15 shows the different phases of the dynamic gesture recognition algorithm, starting from the 3D skeleton captured from the calibrated Kinects to the gesture in output.

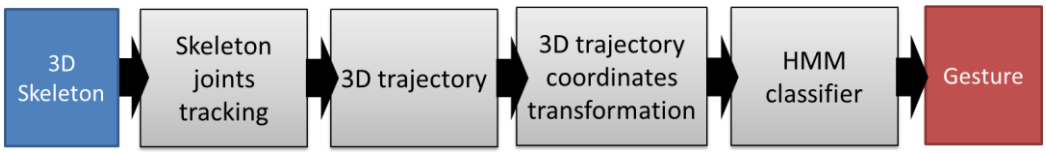


**Figure 4.15 Block diagram representing the dynamic gesture algorithm.**

## 4.3 Test

Two different tests have been conducted. The first test aimed at the evaluation of the gesture recognition accuracy, in order to assess the validity of this approach for the view-invariant gesture recognition of both deictic and dynamic gestures. The second test involved the setup of a daily living scenario in order to understand the usability of such system in a smart living room.

### 4.3.1 Gesture Recognition Test

This test had a twofold aim: firstly, to demonstrate that this non-intrusive 3D gesture recognition approach grants excellent results for both deictic and dynamic gestures; secondly, to demonstrate how the use of two depth cameras can effectively extend the interaction area for view-invariant gesture recognition. The prototype that has been developed as proof of concept has been tested in in two different configurations: the first one used only one Kinect

camera; the second configuration had two calibrated Kinects. The subjects of this test were 10 users (2 women) with different backgrounds and origins, and with age between 19 and 30 years. Every user had to perform the same test procedure in both the configurations and he/she had to take his/her place as showed in Figure 4.16.

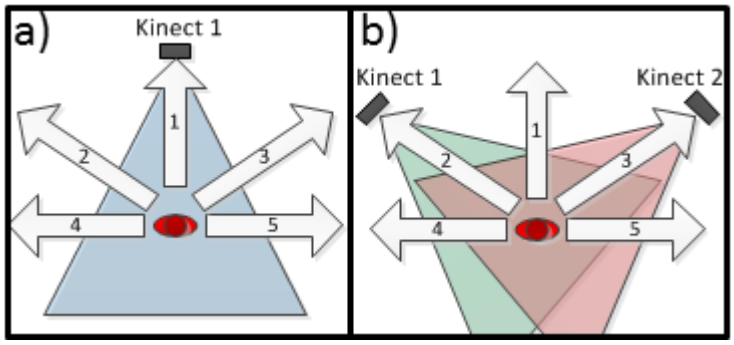


**Figure 4.16 Test configurations and user multi-angle positions: a) the system configuration with one Kinect, b) the system configuration with two Kinects; the red shape is the user.**

In this figure, the arrows indicate the five user's angular positions: position 1 means that the user had to look in the direction of the arrow labeled with the number 1, defined as position 0°. Therefore, the position 2 represents -45° angle, position 3 is +45°, position 4 is -90° and position 5 is +90°. The test procedure consisted of two phases. The first one was the training phase: the user had to perform eleven times the three dynamic gestures. In fact, only the dynamic gesture recognition needs the training phase. The training gestures were performed always in the same angular orientation, precisely the position 1 (a user training the system is shown in Figure 4.17). The first phase recorded data were used to train the HMM classifier. In the second phase, the user had to perform four times each of the three dynamic gestures and four times the pointing gesture in all the aforementioned five angular positions.

**Figure 4.17 One of the users performing a gesture during the tests.**

The three dynamic gestures that were chosen for this evaluation are “left”, “right” and “circle” as shown in Figure 4.18. “Left” gesture started when the user put his/her hands together in front of his/her chest, then he/she extended the left arm horizontally to the side. “Right” gesture started when the user put his/her hands together in front of his/her chest, then he/she extended the right arm horizontally to the side. “Circle” gesture started when the user put his/her hands together in front of his/her chest, then he/she raised the arms up and performed a complete rotation of both arms. These gestures are composed of directional dynamic gestures (the “left” and “right” gestures) and the non-directional “circle” gesture. Hence, it was possible to evaluate the prototype with both types of dynamic gestures, plus the pointing gesture.

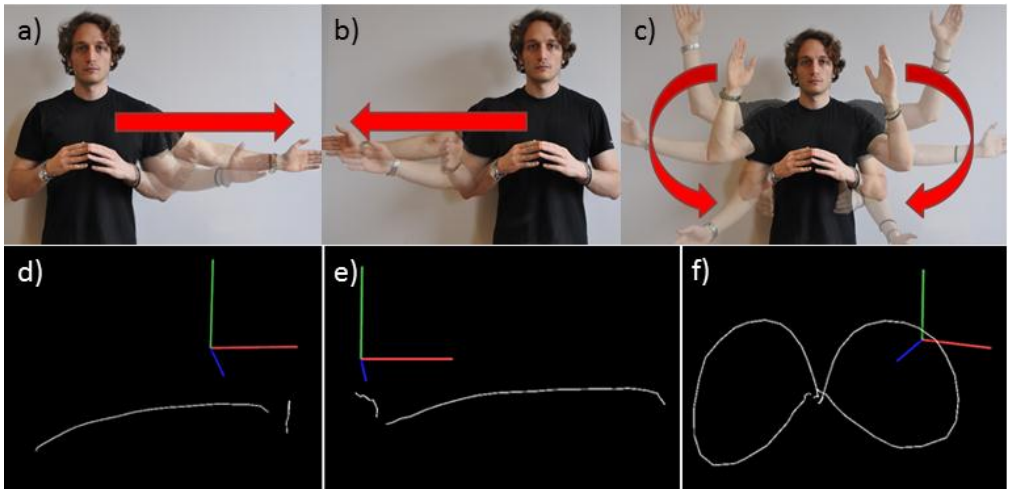


**Figure 4.18 Dynamic gestures chosen for the test: a) "left", b) right and c) circle; d) represents the 3D trajectory associated to the gesture a), e) is associated to b) and f) to c)**

In this test, the 10 users performed a total amount of 2260 gestures, 1130 gestures for each configuration. Therefore, each test procedure involved 1130 gestures of which 330 were for the training phase of the dynamic gestures (the pointing gesture recognition does not require the training phase) and 800 for the recognition phase. The confusion matrix of the evaluation of the configuration with one Kinect is reported in Table 4.1. Table 4.2 reports the confusion matrix of the evaluation of the system configuration with two calibrated Kinects.

**Table 4.1 Confusion matrix for the evaluation of the system configuration with one Kinect; the number of recognized gestures are reported for every gesture and every position, and the relative percentage is between parentheses.**

| 1 Kinect configuration | | | Recognized gesture | | | |
|---|---|---|---|---|---|---|
| | | | Left | Right | Circle | Pointing |
| Performed gesture | Position 1 | Left | 40 (100%) | 0 (0%) | 0 (0%) | 0 (0%) |
| | | Right | 0 (0%) | 40 (100%) | 0 (0%) | 0 (0%) |
| | | Circle | 0 (0%) | 0 (0%) | 40 (100%) | 0 (0%) |
| | | Pointing | 0 (0%) | 0 (0%) | 0 (0%) | 40 (100%) |
| | Position 2 | Left | 36 (90%) | 2 (5%) | 2 (5%) | 0 (0%) |
| | | Right | 0 (0%) | 37 (92.5%) | 3 (7.5%) | 0 (0%) |
| | | Circle | 0 (0%) | 2 (5%) | 38 (95%) | 0 (0%) |
| | | Pointing | 0 (0%) | 0 (0%) | 0 (0%) | 40 (100%) |
| | Position 3 | Left | 37 (92.5%) | 1 (2.5%) | 2 (5%) | 0 (0%) |
| | | Right | 1 (2.5%) | 36 (90%) | 3 (7.5%) | 0 (0%) |
| | | Circle | 1 (2.5%) | 1 (2.5%) | 38 (95%) | 0 (0%) |
| | | Pointing | 0 (0%) | 0 (0%) | 0 (0%) | 40 (100%) |
| | Position 4 | Left | 7 (17.5%) | 27 (67.5%) | 6 (15%) | 0 (0%) |
| | | Right | 2 (5%) | 27 (67.5%) | 11 (27.5%) | 0 (0%) |
| | | Circle | 0 (0%) | 21 (52.5%) | 19 (47.5%) | 0 (0%) |
| | | Pointing | 0 (0%) | 0 (0%) | 0 (0%) | 40 (100%) |
| | Position 5 | Left | 24 (60%) | 7 (17.5%) | 9 (22.5%) | 0 (0%) |
| | | Right | 9 (22.5%) | 21 (52.5%) | 10 (25%) | 0 (0%) |
| | | Circle | 6 (15%) | 21 (52.5%) | 13 (32.5%) | 0 (0%) |
| | | Pointing | 0 (0%) | 0 (0%) | 0 (0%) | 40 (100%) |

**Table 4.2 Confusion matrix for the evaluation of the system configuration with two Kinects; the number of recognized gestures are reported for every gesture and every position, and the relative percentage is between parentheses.**

| 2 Kinects configuration | | | Recognized gesture | | | |
|---|---|---|---|---|---|---|
| | | | Left | Right | Circle | Pointing |
| Performed gesture | Position 1 | Left | 40 (100%) | 0 (0%) | 0 (0%) | 0 (0%) |
| | | Right | 0 (0%) | 40 (100%) | 0 (0%) | 0 (0%) |
| | | Circle | 0 (0%) | 0 (0%) | 40 (100%) | 0 (0%) |
| | | Pointing | 0 (0%) | 0 (0%) | 0 (0%) | 40 (100%) |
| | Position 2 | Left | 39 (97.5%) | 0 (0%) | 1 (2.5%) | 0 (0%) |
| | | Right | 0 (0%) | 39 (97.5%) | 1 (2.5%) | 0 (0%) |
| | | Circle | 0 (0%) | 0 (0%) | 40 (100%) | 0 (0%) |
| | | Pointing | 0 (0%) | 0 (0%) | 0 (0%) | 40 (100%) |
| | Position 3 | Left | 39 (97.5%) | 0 (0%) | 1 (2.5%) | 0 (0%) |
| | | Right | 0 (0%) | 37 (92.5%) | 3 (7.5%) | 0 (0%) |
| | | Circle | 0 (0%) | 0 (0%) | 40 (100%) | 0 (0%) |
| | | Pointing | 0 (0%) | 0 (0%) | 0 (0%) | 40 (100%) |
| | Position 4 | Left | 37 (92.5%) | 0 (0%) | 3 (7.5%) | 0 (0%) |
| | | Right | 0 (0%) | 39 (97.5%) | 1 (2.5%) | 0 (0%) |
| | | Circle | 0 (0%) | 0 (0%) | 40 (100%) | 0 (0%) |
| | | Pointing | 0 (0%) | 0 (0%) | 0 (0%) | 40 (100%) |
| | Position 5 | Left | 40 (100%) | 0 (0%) | 0 (0%) | 0 (0%) |
| | | Right | 0 (0%) | 33 (82.5%) | 7 (17.5%) | 0 (0%) |
| | | Circle | 0 (0%) | 0 (0%) | 40 (100%) | 0 (0%) |
| | | Pointing | 0 (0%) | 0 (0%) | 0 (0%) | 40 (100%) |

The pointing gesture was always recognized with 100% accuracy in all conditions and for both the configurations. In fact, the deictic gesture recognition is based on a deterministic approach. For this reason, it is more interesting to analyze the recognition accuracy concerning the dynamic gestures: “left”, “right” and “circle”. In fact, the dynamic gesture recognition is based on a probabilistic

approach. The overall dynamic gesture recognition rate divided by the five user's positions for both the system configurations is reported in Figure 4.19.

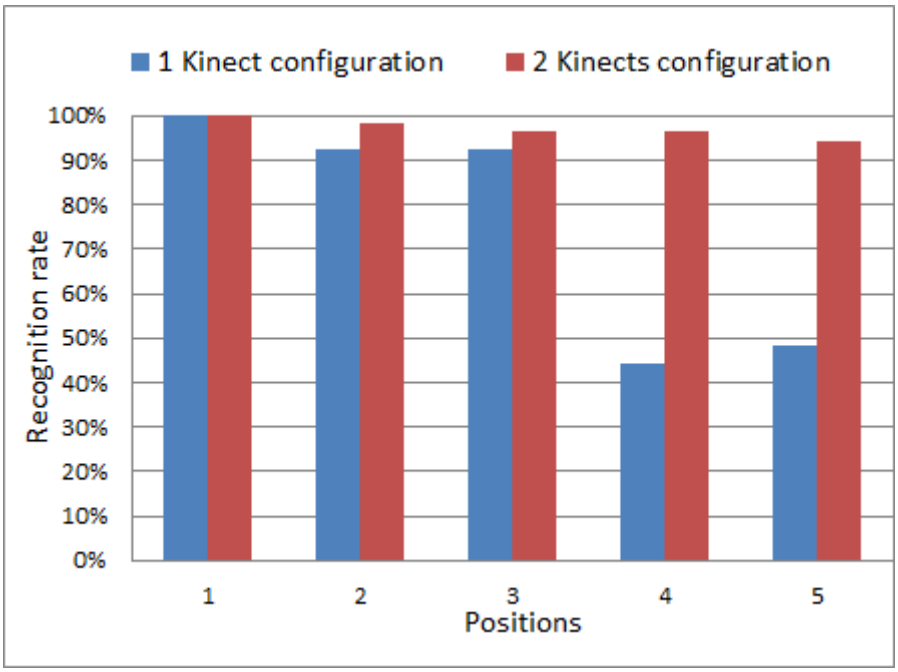


**Figure 4.19 Recognition accuracy rate for only the dynamic gestures expressed in percentage for all the five angular positions and divided by the two system configurations.**

The graphical representation of this graph aims at facilitating the observation of the system performance variation for the probabilistic approach based on the HMMs between the two configurations: with a Kinect and with two calibrated Kinects. In particular, it is possible to notice that during the dynamic gesture recognition of the system configuration with one Kinect, the following recognition rates were obtained: 100% in position 1, 92.5% in position 2 and 3, 44.2% in position 4 and 48.3% in position 5. The same test conducted with the proposed system configuration that integrates two calibrated Kinects gave as results: 100% of gesture recognition rate in the position 1, 98.3% in position 2, 96.7% in position 3 and 4, and 94.2% in position 5. These data confirmed that this non-intrusive 3D trajectory approach with the classification based on HMM provided excellent results for both types of dynamic gestures. Moreover, the system configuration with two calibrated Kinects obtained recognition accuracy

rates for the dynamic gestures that was higher than 90% also in the positions 4 and 5, when the configuration with one Kinect gave rates was lower than 50%. Therefore, using two calibrated Kinects allowed extending the interaction area to an angular orientation of 180°, when the configuration with one Kinect granted good results only for an angular orientation comprised between -45° and +45°.

### 4.3.2 Usability Test

In order to have a feedback about the role of functional gestures and the system usability of this prototype, a usability test composed of two phases was conducted. A setup reproducing a smart living room with four different two-state smart objects was prepared, as the specific scenario presented in section 3.3. These objects are a work lamp, a floor lamp, a fan and a media center. There were two calibrated Kinects for the gesture recognition illuminating the interaction area. The subjects of this test were 13 users (9 men and 4 women) with different backgrounds and origins, and with age between 19 and 28 years.

#### 4.3.2.1 First Phase

The subjects have been conducted to the smart living room where a simple scenario has been prepared. One user at a time has been asked to enter in the room and to interact with the system (see Figure 4.20). The functional gesture allowed switching on and off every smart object. In order to provide the possibility to provide the explicit representation of the user's focus of attention, the functional gesture has been mapped with the pointing gesture.

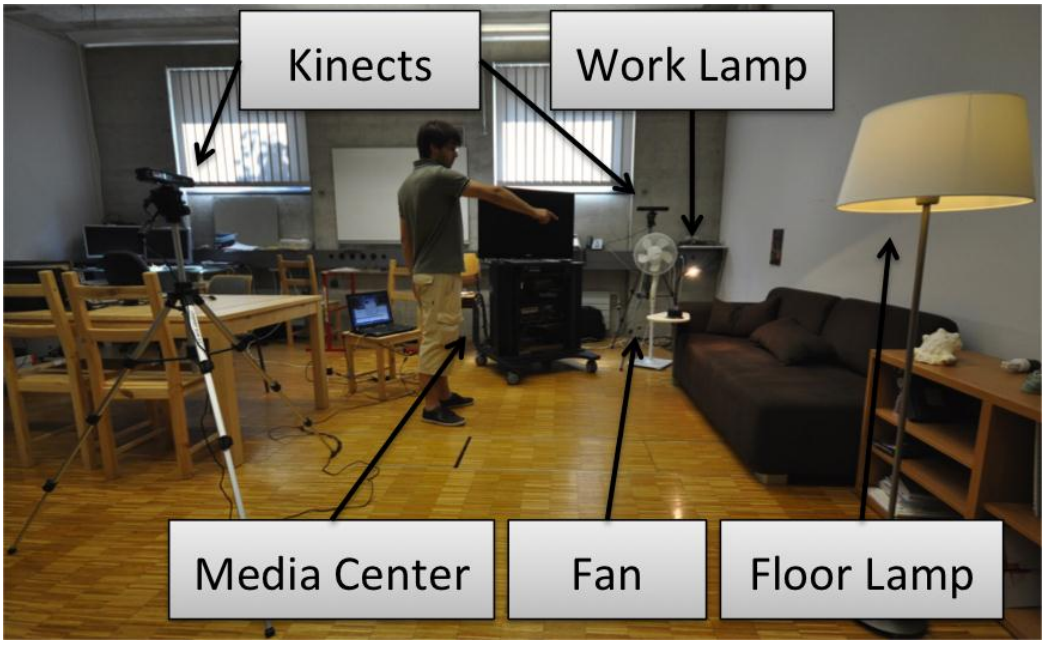


**Figure 4.20 The test scenario.**

After the skeleton tracking initialization stage (the user has to remain in a pose for few seconds in front of each Kinect device), the user had to point at a lamp to turn it on; afterwards the user had to point at the media center to turn on the radio and later he had to do it again to turn it off. Afterwards, he had to sit down on the couch and to point at the media center to turn on the TV (see Figure 4.21). The system never failed the gesture or the posture recognition during the test. Once finished the interaction session in the smart living room, every subject evaluated the experience through the System Usability Scale (SUS) questionnaire rating the system features according to a 5-point Likert scale (Brooke, 1996). The statements covered a variety of aspects of system usability, such as the need for support, training, complexity, efficiency (how much effort is necessary in achieving those objectives) and experience satisfaction. The users' evaluations assessed the system usability as excellent with an average SUS score of 90.6 points and a standard variation of 5 points.

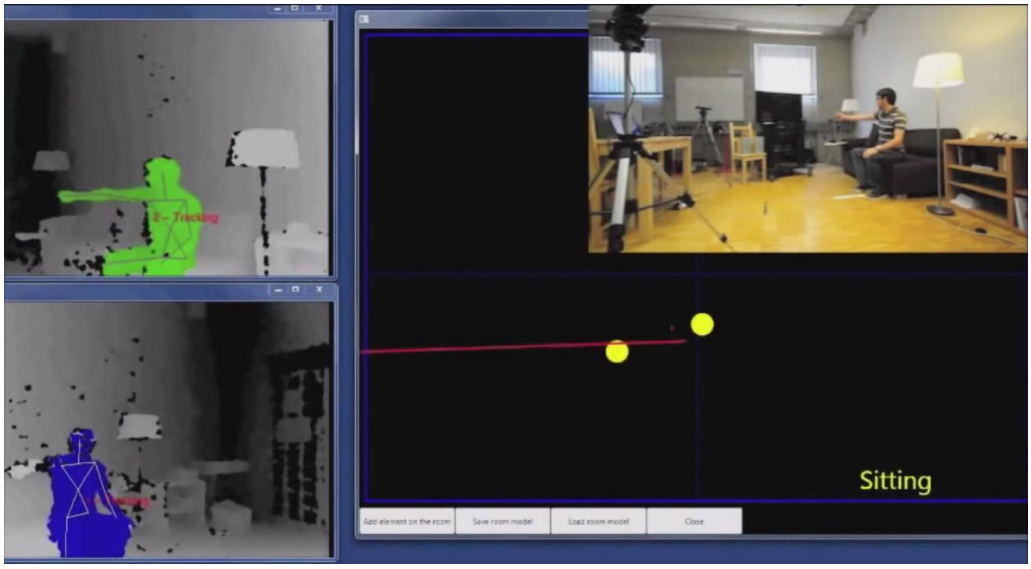


**Figure 4.21 The test scenario with the information elaborated by the system.**

#### 4.3.2.2 Second Phase

This phase consisted in an interview where the users have been asked to express their impressions and suggestions. Most of them said that the skeleton tracking initialization stage could be really annoying for an everyday interaction in a real smart room. The subjects have been asked to say if they have missed the voice interaction modality in this test scenario and everybody answered negatively, moreover they expressed their appreciation about this interaction modality through deictic gestures. The principle of functional gesture did not create problems for the testers, there was not confusion and the test subjects stated their preference in having only one gesture instead of eight or two. In fact, if the functional gesture approach was not adopted, using the direct mapping between the commands and gesture would require eight different gestures (for four different smart objects with two states); the object-aware system would involve the use of two different gestures, one per state. The functional gesture allowed using only the pointing gesture to select the object and the context information permitted to opportunely switch on or off the selected object.

Some of the users remarked that they would like also other gestures to go beyond the turning on or off the household appliances, e.g., they would like to interact with the media center to change TV program or the volume.

Another limitation that came from this analysis is the pre-determined set of tasks that the system executes referring to the users' gestures and postures. In fact, a system that automatically learns user's habits could be preferable to a programmed one. Therefore, in order to make this system more human-centered, the integration of activity recognition and learning algorithms has been considered for future developments in order that the system can learn users' habits.

## 4.4 Applications

The user-centered design approach is an iterative process. The tests presented in the previous paragraph are only the final stage of the first complete prototype. In this paragraph, the gestural interface is adapted to new applications addressing the accessibility issue in ubiquitous computing era. In these applications, the adapted interfaces are tested with a group of people belonging to the target users.

### 4.4.1 Accessibility for Disabilities

The prototype of the view-invariant 3D gesture recognition can offer other possibilities for the enhancement of everyday life. In particular, a gestural interface that allows interacting from a certain distance with smart objects can facilitate the life of people with mobility problems. In fact, reaching a physical switch on the household appliances is a quite difficult and tiring task for physically impaired people on wheelchair. Therefore, the gesture recognition prototype has been adapted to grant accessibility to the control of household appliances for the mobility-impaired people. This prototype is based on a new device, called "BTSwitch", enabling direct remote control of multiple electric plugs. Users can interact with devices in the environment alternatively using a smartphone or the using pointing gesture, as shown in Figure 4.22. The smartphone has a rich user interface, whereas the pointing gesture is based on the natural interaction paradigm.

**Figure 4.22 On the left: the user interacting with the system through the smartphone; on the right: the user interacting with the system through gestures.**

The system is presented in Figure 4.23. A user on a wheelchair can control his/her surrounding environment using one of the two paradigms: using simple touch gestures on a smartphone or using natural interaction by pointing at a device. With the smartphone paradigm, the information is sent directly to the BTSwitch from the smartphone and the user has a direct, rich feedback on his screen. With the natural pointing, a specific computer tracks the user through a Microsoft Kinect camera adopting the approach proposed in this thesis. When a command is detected, it is sent to the BTSwitch and the user receives an acoustic feedback. Dumb appliances are directly plugged into the BTSwitch.

Both interaction paradigms can be used separately or simultaneously according to the user preferences and to the availability of each device: the smartphone paradigm involves the presence of the handheld device; the natural interaction paradigm is limited to the Kinect camera field of view.

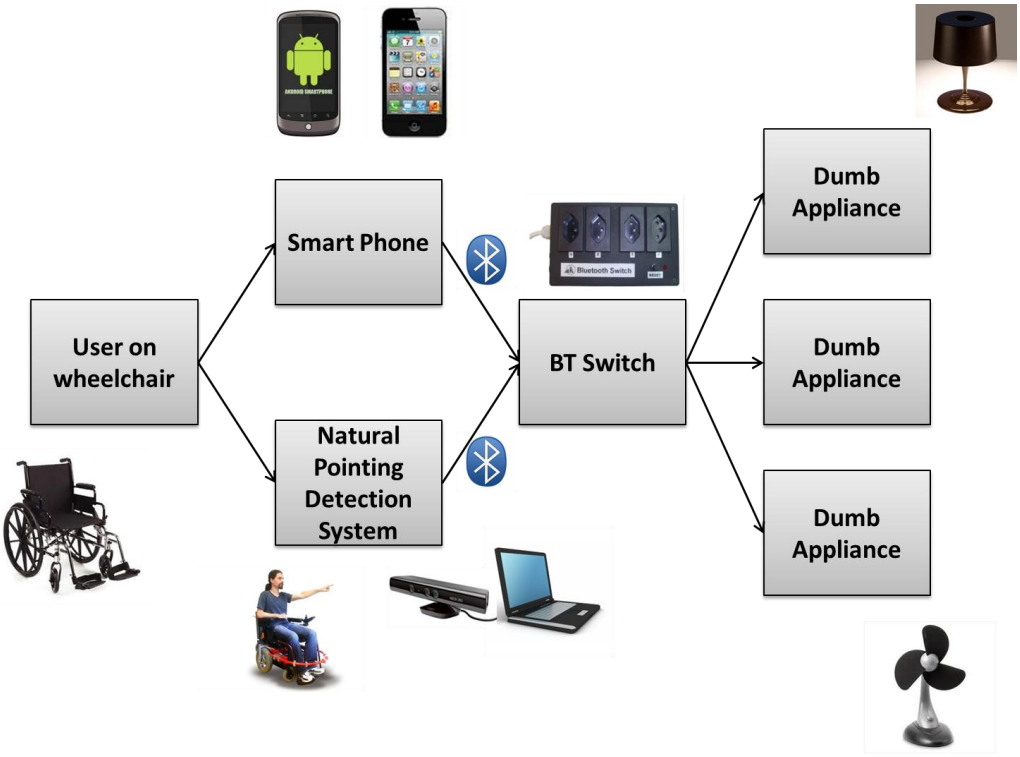


**Figure 4.23 Overview of the system.**

#### 4.4.1.1 Hardware

The BTSwitch power strip prototype has been developed at the Univesity of Applied Sciences and Arts Western Switzerland in Fribourg with the aim of offering a simple and low cost solution to control dumb appliances while providing a plug-and-play installation system when used in conjunction with most smartphones available on the market. To fulfill these requirements, the choice of protocol has rapidly been oriented toward Bluetooth, being widely available on most smartphones and personal computers. This protocol also provides an interesting limitation: its medium range, which provides an implicit approximate localization of the controlled devices.

The BTSwitch power strip prototype is shown in Figure 4.24; it controls up to four plugs and turns them on or off using mechanical relays. The electronic board is directly powered through the current used to supply the plugs. A LED indicates when remote communication occurs and a physical button provides the possibility to turn all plugs off manually.

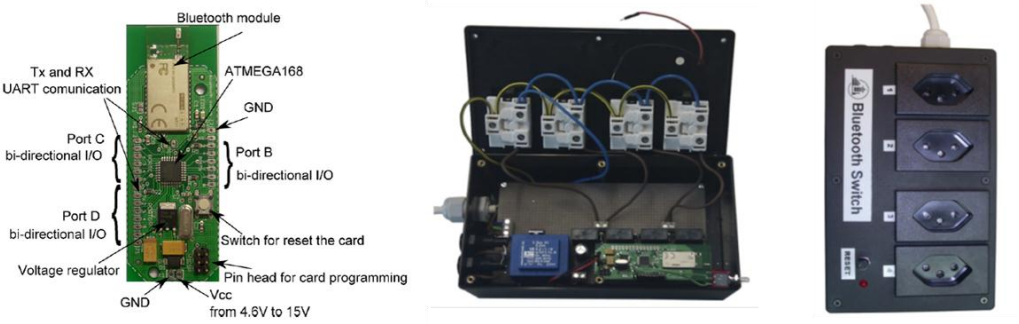


**Figure 4.24 Prototype of the Bluetooth switch: on the left, the custom electronic board and on the right the final prototype.**

#### 4.4.1.2 Interfaces

A first interface is the smartphone application that works as a universal remote controller. This application automatically discovers the surrounding BTSwitch modules and displays the discovered appliances. Within the application, the user can configure each plug and power strip with custom names and images. The display and configuration interfaces for the Android platform are illustrated in Figure 4.25. As shown in the screenshot on the right in Figure 4.25, the graphical interface can be customized according to user preference with a particular focus on size and position of the buttons. On the main interface of the application (Figure 4.25, left), each button has different colors to indicate its state; the green color indicates that the plug is on; the red color indicates that the plug is off and the orange color indicates the transition while the message is being processed. Note that in normal conditions, the time to process a message is less than two hundred milliseconds. To control an appliance, the user simply selects the desired BTSwitch power strip; then, the corresponding appliances are shown in the interface. The desired appliance is turned on or off in real-time by tapping on the corresponding button.

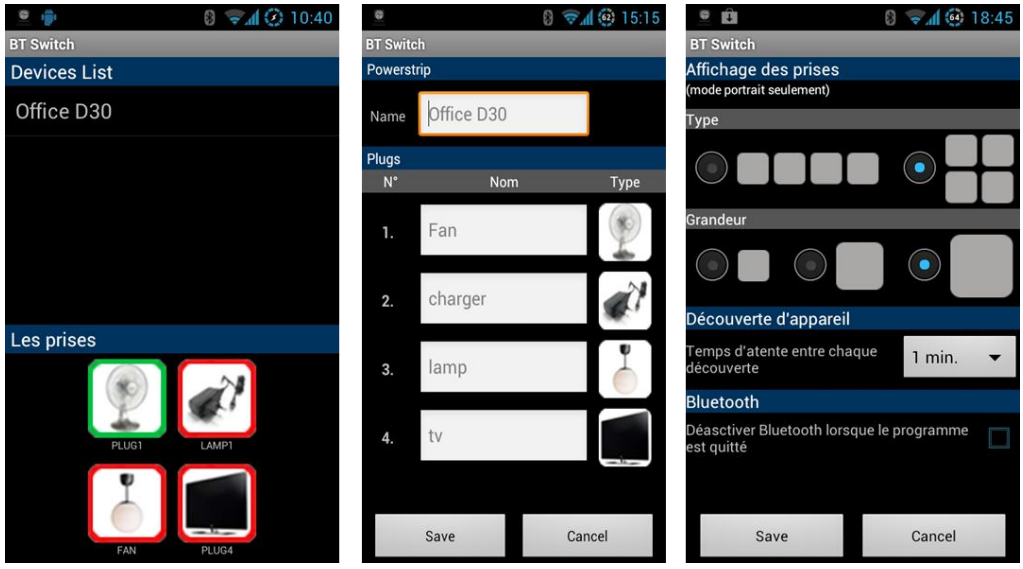


**Figure 4.25 The interfaces of the smartphone application: on the left, the main interface to interact with the ap- pliances; in the center the interface to configure a particular power strip; on the right, the interface application configuration.**

Along with the interaction performed through the smartphone, the natural interface based on the functional gesture using the pointing gesture as presented in the previous section has been implemented. The only difference with the approach of the previous test is the number of tracked joints: in order to optimize the user tracking for people in a sitting position (i.e., people on a wheel-chair), the joints of the legs were filtered. The spatial model for the context information and the gesture detection was reconstructed also in a 3D graphical representation as depicted in Figure 4.26. The 3D axes represent the user's arm with the yellow extension for the calculation of the target object. In this case, the smart objects were modeled as cubes.

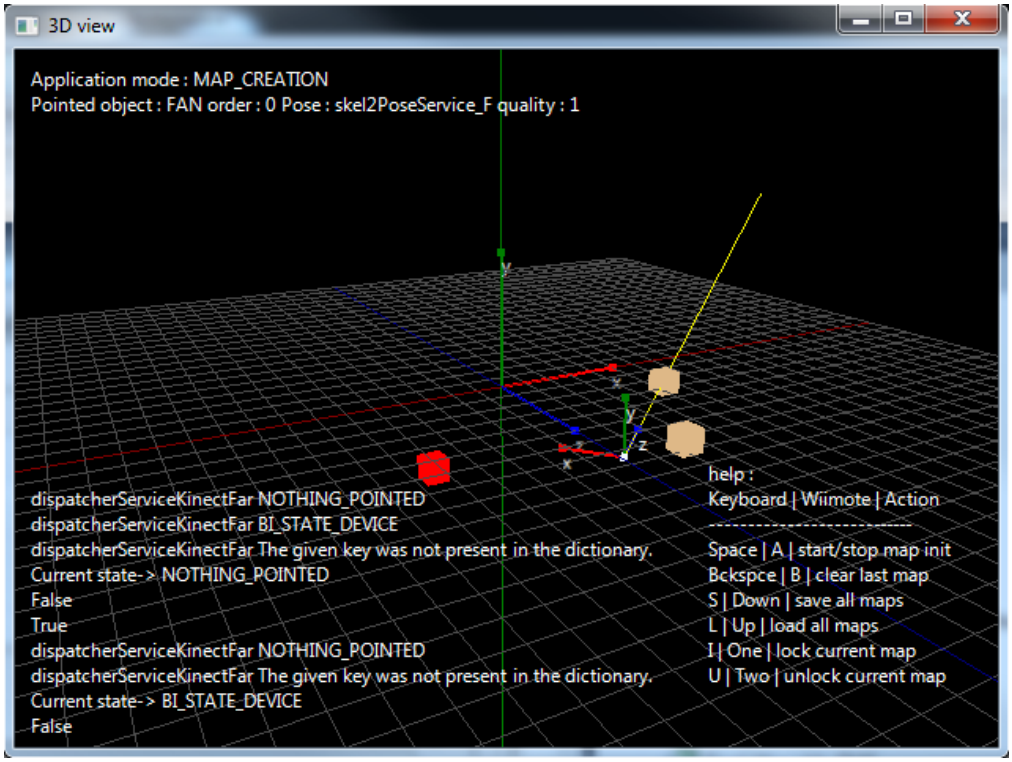


**Figure 4.26 The 3D graphical representation of the pointing gesture recognition; the red line is the ray and the cubes are the smart objects.**

#### 4.4.1.3 Usability Test

The scenario was set up in an office context as shown in Figure 4.27. The user is the first person entering in his/her work office; therefore, he/she has to turn the light on, to power on his/her personal computer and to start working. After a while, the office temperature raises and the user decides to turn the fan on for some freshness. He/she continues working for some time, then, before leaving his/her office, he/she turns all the electrical appliances off.

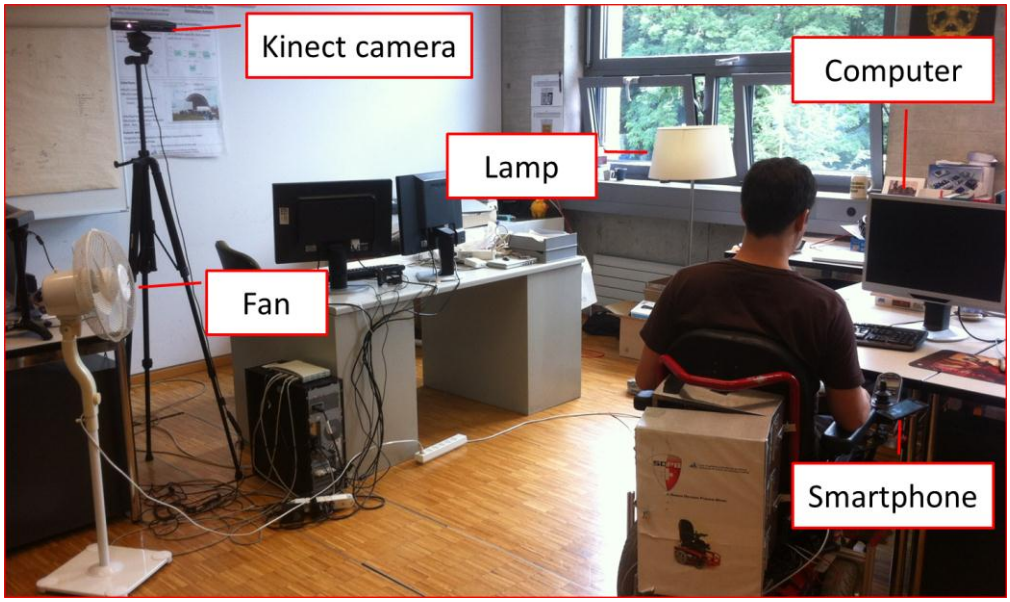


**Figure 4.27 Test scenario.**

The system was tested with 13 users (1 person with reduced mobility on his own wheelchair and 12 able-bodied subjects on an electric wheelchair) with age ranging from 21 to 34. The experiment was composed of two phases; one phase involved the user following the office scenario controlling the electrical appliances through the smartphone paradigm. The other phase consisted in accomplishing the same scenario controlling the electrical appliances through the natural interaction paradigm. The order of the two phases was randomly chosen for each subject. After each phase the subject had to fill in the SUS questionnaire (Brooke, 1996). At the end of the whole experiment the subject filled in a questionnaire with five open questions: which interaction paradigm he/she preferred and the advantages and the disadvantages of each interaction paradigm.

The users' evaluations assessed the smartphone interaction paradigm usability as excellent with an average SUS score of 91.3 points and a standard deviation of 7.5 points. The system with the natural interaction paradigm usability obtained an average SUS score of 84.2 with a standard deviation of 7.6.

According to users' feedback written in the questionnaires, the main advantages of the smartphone interaction paradigm are that it is reliable, intuitive, requires minimal effort, and that all the controllable appliances are visible on the screen

with direct visual feedback of their state. The main disadvantages have been identified as the required precision to press a button on the touch screen and the need of carrying a handheld device. The main advantages of the natural interaction paradigm are that it is very intuitive, provides a natural interaction mechanism (absence of handheld device) and a direct visual feedback from the physical appliances. The main disadvantages are the need to move the wheelchair to control specific appliances, and the potential fatigue caused by the deictic gestures.

The subject with impaired mobility preferred the smartphone interaction paradigm. He specially emphasized the fact that most electric wheelchair users already have a smartphone attached to their wheelchair and the convenience of such a system for people with reduced mobility of the upper limbs. On the other hand, he also identified the advantage of the natural interaction paradigm for people with reduced mobility of the fingers that could find the smartphone interaction more troublesome. He also stated that, for both cases, a vocal modality could be a great additional feature.

### 4.4.2 Accessibility for Democratic Design and Development

Developing vision-based 3D gestures recognition systems requires strong expertise and knowledge in computer vision and machine learning techniques. Human-computer interaction researchers do not generally have a thorough knowledge of these techniques; in fact, HCI researchers can have different backgrounds in very different domains such as computer science, sociology, psychology, communication, human-factors engineering, industrial engineering, rehabilitation engineering, and many others. Hence, many HCI researchers do not have a thorough knowledge of all the techniques belonging to these fields. In order to enable everyone to conduct research for 3D gestural interfaces, a tool based on the proposed approach of 3D gesture recognition was developed. This full-fledged tool enables non-experts in vision computing and machine learning techniques to rapidly develop a prototype for 3D dynamic gesture recognition.

This tool permits to manage up to two Microsoft Kinect cameras. The architecture is composed of many modules as shown in Figure 4.28 a).

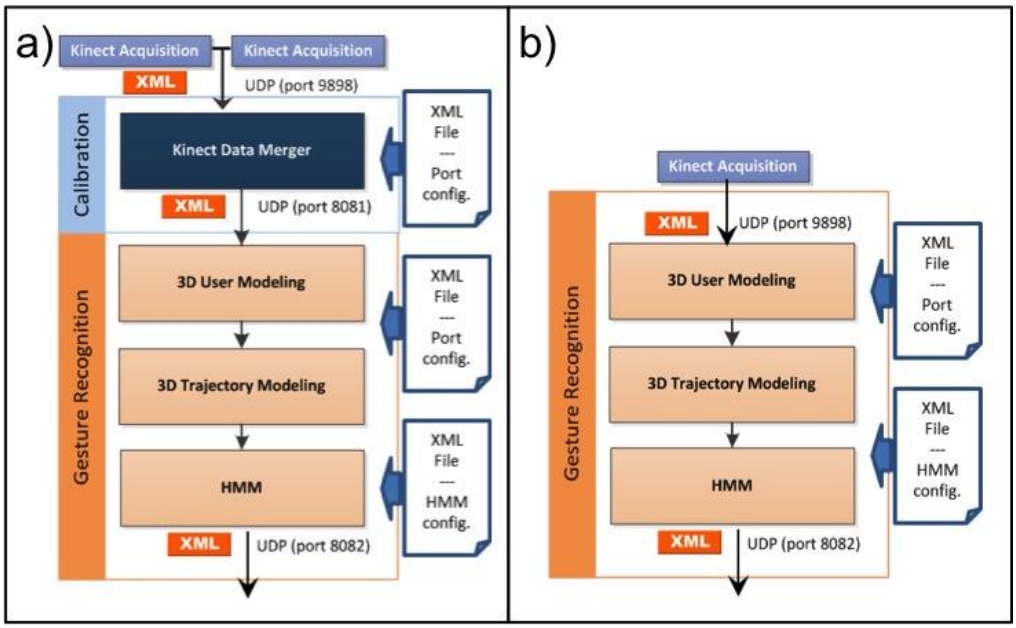


**Figure 4.28 The two configurations of the tool architecture: on the left with two Kinects, on the right with a single Kinect.**

Two modules in C++ and based on the OpenNI libraries are dedicated to manage the connected Microsoft Kinects. Each “Kinect Acquisition” module manages a single Kinect. These modules track the person in the field of view and construct the associated skeleton. These two modules send the two skeleton models to the “Kinect data Merger” module in a specific XML message via UDP. Every XML message contains the information about the coordinates of every skeleton joint and the ID number of the Kinect camera that sent these data. The “Kinect Data Merger” module is dedicated to the fusion of the two skeletons provided by the “Kinect Acquisition” modules. Once the two 3D skeletons are calculated, this module makes the fusion of the data concerning the tracked person in the interactive area to create a unique 3D skeleton. The joints that are considered during the fusion process must have the maximum value of the associated reliability factor that is provided by the OpenNI libraries. A specific GUI is associated to this module in order to allow the user to calibrate the two Kinects

with a few number of mouse clicks. The algorithm used for this calibration will be briefly descripted in the next section. The “Kinect Data Merger” sends the transformed coordinates to the “Gesture Recognition” block. This functional block is composed of three different modules. The first one reconstructs the 3D skeleton model as a half stickman composed of the following joints: head, neck, right and left shoulders, right and left elbows, right and left hands, torso, right and left hips. The second module tracks the selected joints and calculates the relative 3D trajectories in a space reference frame that has origin between the skeleton shoulders. The 3D trajectories are sent to the “HMM” module. This module integrates a hidden Markov model (HMM) classifier that, once trained, recognizes the captured 3D gestures. Afterwards, the “HMM” module sends every recognized gesture in a XML message via UDP.

This modular architecture allows two configurations: with one or two Kinects. The effective system architecture in configuration with only one Kinect is depicted in Figure 4.28 b). Moreover, this modular architecture grants future upgrades that should allow the connection of up to four Kinects for the view-invariant gesture recognition on 360°. In addition, this modular structure enables the programmers to reuse the functional blocks in their systems making them to save time and energy.

Some parameters must be set in the XML files. In fact, the UDP ports of input and output of the “Calibration” block must be configured in a specific XML file (the default ports are shown in the schemas). Similarly, the input and output ports of the “Gesture Recognition” block are configured in another XML file. This block has also an XML file for the configuration of the HMM parameters. In particular the number of hidden states and the typology can be specified in this XML file. The default HMM classifier has four hidden states, is ergodic and uses the Baum-Welch algorithm to find the unknown parameters.

This tool boasts other two important characteristics: firstly, the whole system is very inexpensive (since it requires only two Kinects and these device are quite

cheap); secondly, if it is installed in a laptop and the Kinects are mounted on adjustable tripods, the whole system becomes very easy to transport.

#### 4.4.2.1 User Interface

This tool provides a specific GUI that allows the user to easily access and manage complex functions such as the Kinects calibration and the generation of an HMM classifier. The GUI is composed of three different environments: the first one is dedicated to the calibration between the two Kinects; the second one handles the recording of the 3D gestures and the generation of an HMM classifier; the last one is a multimedia documents manager for advanced gesture analysis.

The GUI for the calibration allows displaying the information about the two connected Kinects (Figure 4.29).

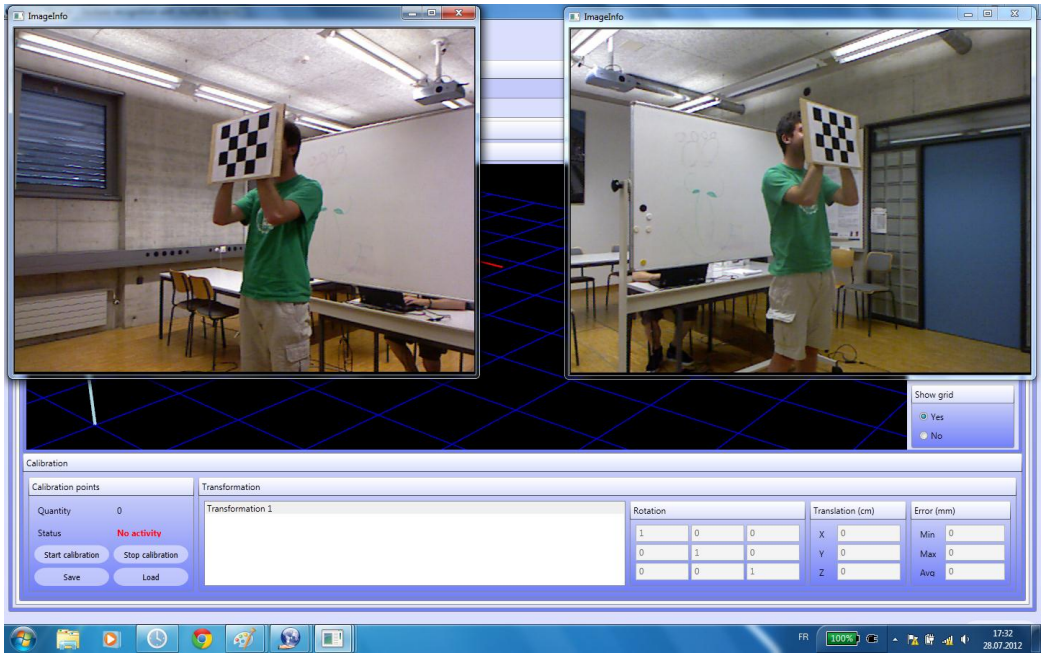


**Figure 4.29 The tool interface during the calibration phase.**

There are two tabs that show the 3D information coming from each Kinect as represented in Figure 4.30. In these tabs, it is possible to start the points acquisition process. That means that once the user has positioned the checkerboard in front of the Kinects, the system automatically captures the 3D point sets and synchronizes them.

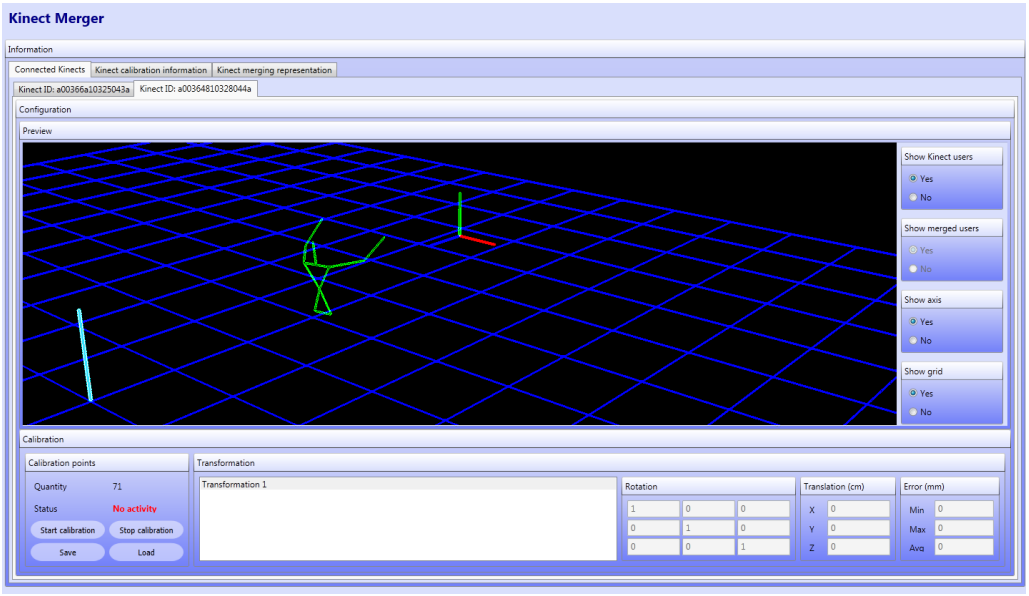


**Figure 4.30 Visualization of the 3D skeleton provided by one Kinect: user's skeleton model (green) with the floor (blue), the normal to the floor (white) and the reference frame axes.**

The user can save the calculated data. Afterwards, the user has to choose the Kinect that he/she wants to use as reference system; then, the user has to click on a button and the tool calculates the transformation matrices. These matrices can be saved; this means that if the user does not move the Kinects, he/she can reuse the calculated transformation matrices to recalibrate the Kinects. The matrices will be displayed in the Kinect tab with the relative error. In another tab, it is possible to visualize the two 3D skeletons captured by the Kinects and the merged skeleton together. These skeletons can be distinguished thanks to the different colors and the labels. This part of the GUI is depicted in Figure 4.31, but it provides also some buttons so the user can choose the skeletons to display in real time.

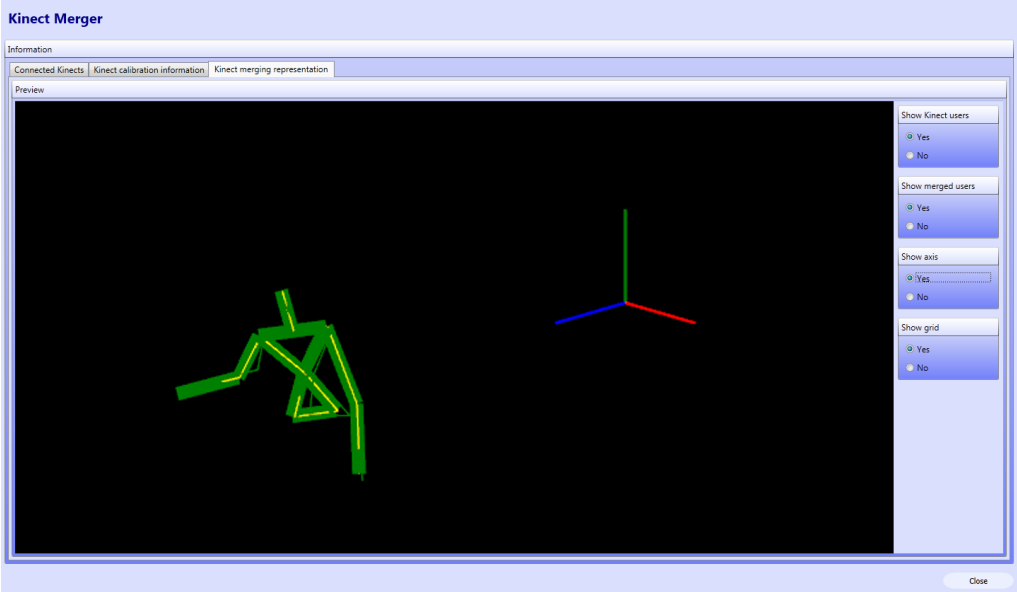


**Figure 4.31 Visualization of the 3D skeleton after the merging phase: the merged skeleton model is the one with thick green lines; the skeletons with thin lines are the ones captured by the Kinects.**

The second interface environment is for the gesture recognition. The tool allows adding new gesture categories and to record several gestures for each category. Moreover, the user has to choose the 3D skeleton joints that he/she wants to track in order to define the gestures (Figure 4.32).

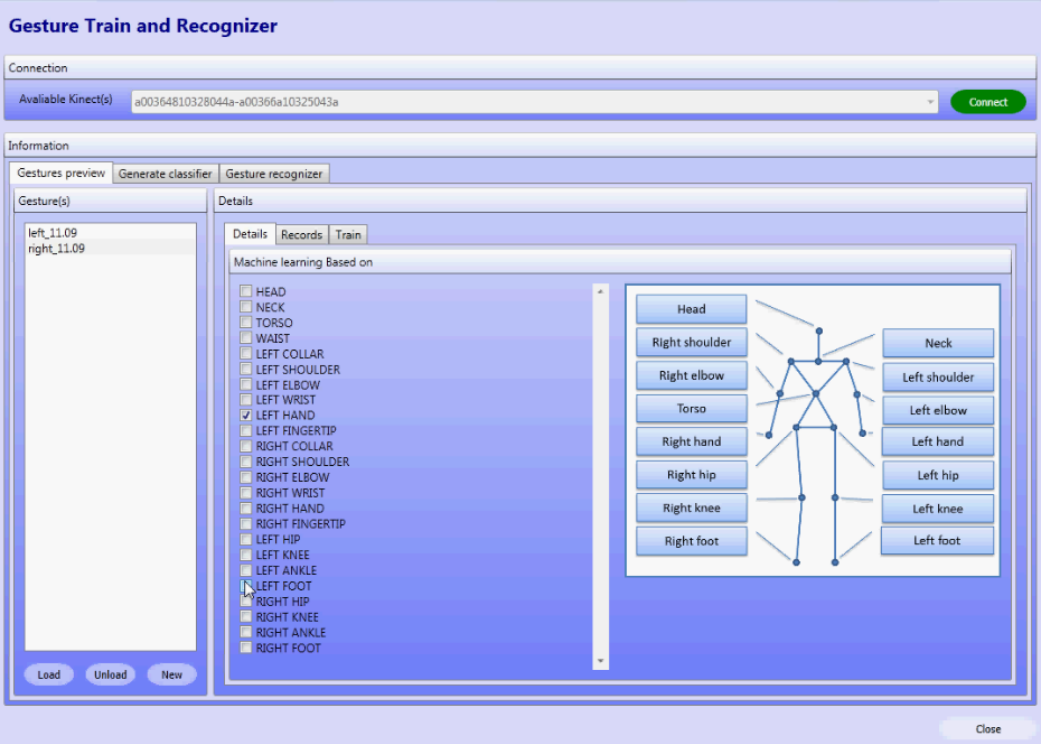


**Figure 4.32 The tool interface for the selection of the joints to track for the gesture training and recognition.**

The GUI allows visualizing the recorder data concerning every record. These data are shown in a 3D space that can be zoomed and rotated using the mouse as depicted in Figure 4.33; in addition, the user can press the play button and the tool reproduces the gesture data as an animation. The 3D representation of the gestures data aims enhancing the design process. The gesture categories and the records are saved in the computer and the user can load, unload and modify them in the tool without constraints. The freedom of configuration provided by this tool enhances the support of iteration and retrospection during the gesture design process. The developed tool offers also the possibility to use the loaded records to train an HMM classifier in order to obtain a functioning gesture recognition system prototype. The user has only to push a specific button to generate this classifier but in this case there is a constraint: the loaded records must be registered tracking the same joints. The gesture segmentation can be determined manually by the user pressing the specific buttons or activating the automatic segmentation (start: joined hands, stop: unmoving tracked joints).

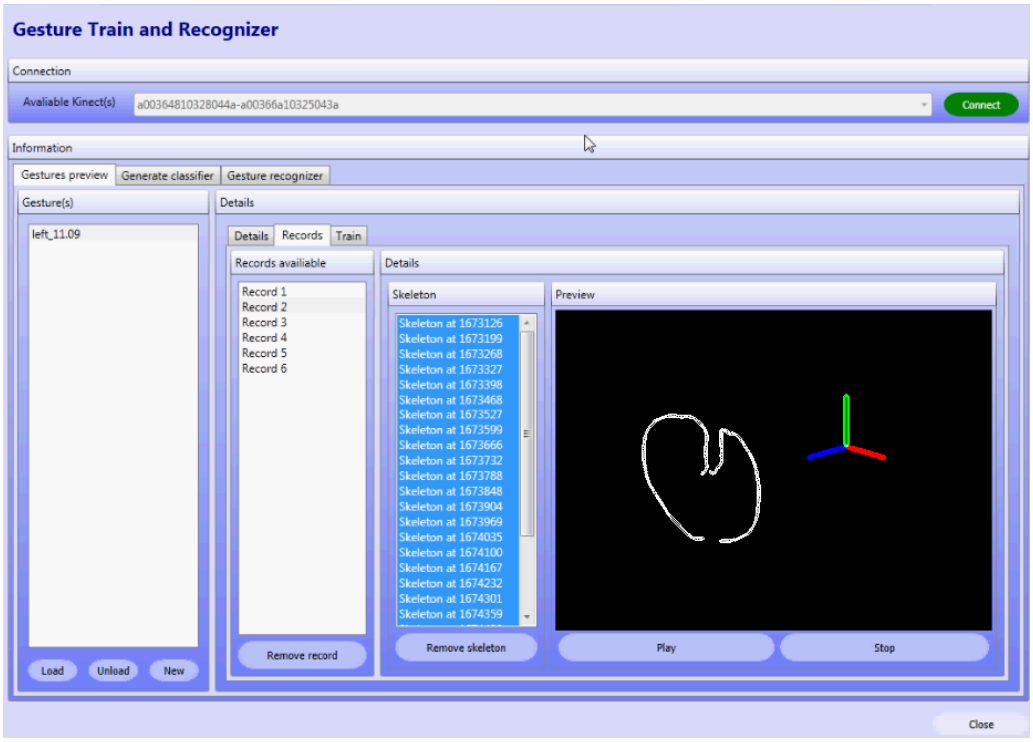


**Figure 4.33 Interface for the visualization of the 3D trajectories corresponding to the recorded gestures.**

The last GUI environment has been designed to support further testing of the designed gestures. This part of the tool allows the user to manage multimedia documents in order to facilitate the study of learnability, social acceptability, usability in different contexts et cetera. In fact, comprehensive gesture analyses can be dispersive: managing a huge amount of data can become hard, even more so if they are composed of different types and formats. This tool allows regrouping and managing all the information concerning a recorded gesture; in addition, it supports several formats and permits displaying pictures and videos directly in the GUI (Figure 4.34).

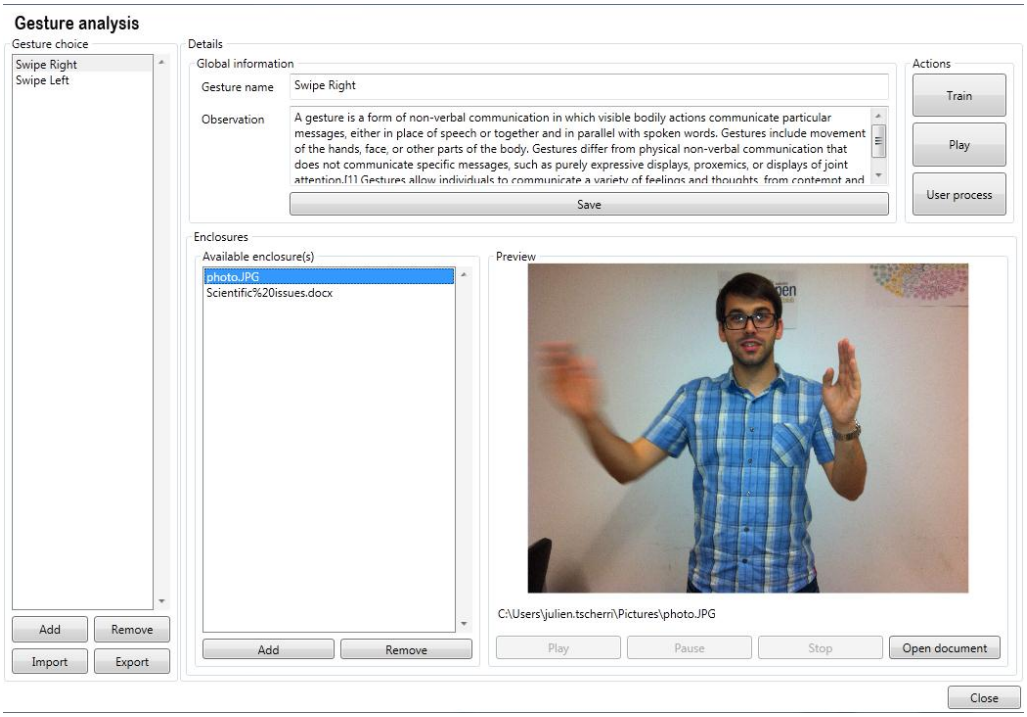


**Figure 4.34 Interface for the management of multimedia documents for gesture analysis.**

#### 4.4.2.2 Usability Test

In order to evaluate the usability of this tool, a test with 12 users (1 woman, age between 22 and 35 years) was conducted. The test subjects are researchers with experience in the HCI domain. The participants were asked to complete a brief demographic survey before starting the experiment. They had to evaluate their experience with designing user interfaces, with computer vision and with machine learning techniques on a 9-point scale with 1 being no experience and 9 being very experienced. Here the demographic information is reported: 7.6 for the experience with designing user interfaces, 4 for the computer vision and 4.8 for the machine learning.

The testers interacted singularly with the system and they have been showed the functions of the system (see Figure 4.35). Then, they were asked to accomplish three tasks: calibrating the Kinects, designing three gestures of their choice and to generate the HMM classifier. A collaborator was in the interactive area in

order to perform the gestures asked by the tester. Once finished the use session, every subject evaluated the tool through SUS questionnaire rating the system features according to a 5-point Likert scale (Brooke, 1996). After the questionnaire, a brief interview with the tester was held.

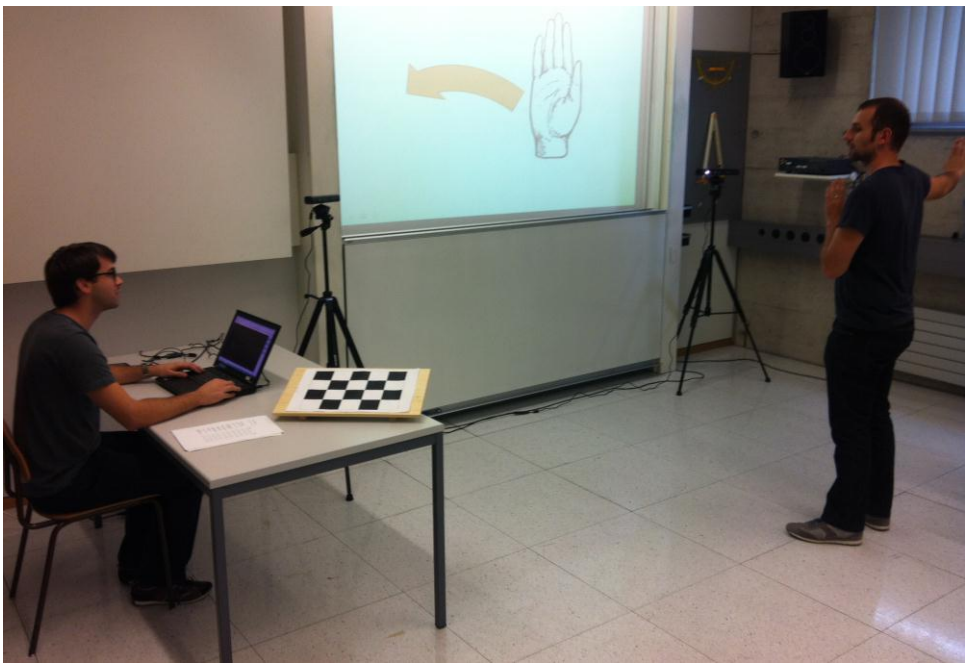

**Figure 4.35 Two testers using the tool during the evaluation phase.**

All the test subjects greatly succeeded in accomplishing the three tasks. The overall SUS following the standard procedure scored 88.9 points out of 100 (standard deviation: 7.2). Moreover, two additional factors were calculated from these data as suggested in (Lewis & Sauro, 2009). The two additional factors are the perceived usability and the perceived learnability, the latter is very important to provide a better understanding of how easy is for HCI researchers to learn to use this system. This tool scored respectively 87.8 out of 100 for the perceived usability (standard deviation: 7.2) and 93.7 out of 100 for the perceived learnability (standard deviation: 8.4).

After the questionnaire, a brief interview with the testers was conducted, which were asked to express their impressions and suggestions. Their impressions were generally positive, in particular referring to the intuitiveness of the GUI; this confirms the good rate scored with the SUS questionnaire. Moreover, the test

subjects provided good suggestions; in particular, they highlighted that showing the information about the two Kinects in the same tab would be preferable. Another improvement that they suggested concerns the automatic gesture segmentation. The testers suggested that adding the possibility of personalization and configuration of the automatic 3D gestures segmentation would be a valuable feature.

## 4.5 Summary

The main contribution of this chapter consists of a novel technique for view-invariant 3D gesture recognition. This technique allows for unobtrusive gesture detection granting the freedom of movements in the environment, which is very important in the human-environment interaction scenario. In particular, two algorithms were developed: one for the pointing gesture and one for the dynamic gesture recognition. The tests assessed the recognition accuracy of this novel approach with good results. Moreover, a usability test has been conducted in order to assess the suitability of an interface developed following the framework and the functional gesture method proposed in this thesis for the human-environment interaction.

The last part of this chapter has been dedicated to the applications of these concepts to real life scenarios with special regard to the accessibility issues. In the first application, the accessibility is intended as physical possibility to access the control of household appliances. For people with mobility impairment, accessing to the physical switches cannot be taken as granted and the gestural interface presented in this thesis allows them interacting with these appliances in a more comfortable way. The second application addresses accessibility intended as the possibility of designing a developing a gestural interface. Due to the intrinsic inter-disciplinary nature of this field, people working in HCI have very different backgrounds, competences and research methods. Often HCI researchers are not domain experts in computer vision or artificial intelligence. For this reason, a tool with an intuitive GUI has been developed in order to grant

to all HCI researchers the accessibility to view-invariant 3D gesture recognition technology. Usability tests and interviews showed that this could help the design and development of 3D gestural interfaces of the future.

# Chapter 5: User Status: Activity Recognition

## 5.1 Introduction

The framework for context-aware gestural interfaces presented in the previous chapter introduces six dimensions for the description of the users and smart objects that are present in the environment. These dimensions are fundamental to represent the relation between them in order to provide an opportune description of the interaction. This thesis focuses particularly on three aspects of the context: the user status, the smart objects status (which constitute the system) and the spatial relation. The spatial relation has been described in the previous chapters with reference to the concept of functional gesture. In this chapter, the dimension of the user status is further analyzed; in particular, the status for a human being cannot be described if the information about the activity is not included. Human activity recognition is a technical challenge with many application areas in HCI, such as context-aware computing. One of the main goals is to recognize activities in order to enhance the interaction between the user and a smart environment that can adapt itself according to the user activities and needs. The activities monitored by the system can be very simple or more and more complex making necessary the presence of several sensors on the person and in the environment. In general, recognizing the user activity provides an important tool to improve and make more efficient the system response to the user needs. For instance, a smart home can adapt several parameters to the user needs automatically, depending from the ongoing activity (e.g., the room temperature, the music volume et cetera). The approach

presented in this thesis is based on the analysis of surface electromyographic (sEMG) signals that measure the electrical activity produced by skeletal muscles during intentional movements; this clinical use of sEMG is often referred to as kinesiologic EMG. The sEMG is the most common non-invasive approach to measure the muscle activity. Using sEMG-based systems in activity recognition provides a main advantage, which is the possibility to recognize subtle or motionless movements that are very difficult or impossible to detect using other more popular technologies, such as inertial sensors. The application of sEMG technology in HCI is not unprecedented but it is a novelty for the ubiquitous activity recognition. In fact, the sEMG has been commonly used for clinical investigations, for example for in-depth analyses of gait-related postural control mechanisms or the estimate of the muscle fatigue status. Then, some pioneering works presented the use of EMG based technology outside the clinical laboratories. One of the very early works was conducted in (Saponas et al., 2008), where the authors presented an EMG-based interface for hand gesture recognition. A later work introduced the possibility of using the muscle activity sensing for the activity recognition as in (Gang et al., 2012). However, this kind of systems used to be applied only to the upper limbs and trunk activities detection. A more interesting example can be found in Chen et al.'s work presented in (Chen et al., 2011). This system aimed to provide empirical stride estimation for pedestrian dead reckoning. However, as already stated, no works dealt with the human activity recognition using only the sEMG. In this thesis, a novel technique for sEMG-based human activity recognition is presented.

## 5.2 Design and development of the prototype

In order to sense the sEMG signals, it is necessary to attach some electrodes directly on the skin in the region corresponding to the muscles, or group of muscles, of interest. The electrode is a sensor that allows recording the muscular activity, and can be defined as a transducer of the ionic current that flows in the tissue to an electrical current that flows into the wire. Therefore, the electrode

placement is crucial in order to detect the meaningful characteristics of the EMG signal and to achieve high accuracy rate in activity recognition. It is necessary to use at least two electrodes for each muscle to sense and the inter-electrode distance between these two sensors is one of the most important constraint of the positioning. In fact, the best practice suggested from the SENIAM european project is to fix the inter-electrode distance at 20 mm independently from the type of electrode (bipolar or array); this distance should be used for the positioning of every pair of electrodes on each muscle to be sensed, independently from the its size (Hermens et al., 1999). In order to simplfy the positioning of electrodes, some companies released a particular type of sensor that integrates a pair of electrodes at the right distance (see Figure 5.1).

**Figure 5.1 The electrode used to sense the EMG signals.**

For the electrode positioning, it is important to consider the movements that have to be recognized. In this experiment, the target activities were “walking”, “running”, “cycling”, “sitting” and “standing”. All these activities are categorized as complex movements meaning that they concern different musculo-scheletric systems. For this reason, the muscles have been selected with reference to two important characteristics: an adequate value of amplitude acquirable using the sEMG and significant differences in the temporal activation among the activities. In the sitting and standing activities, there are not evident changes of the body

position, differently from the other activities. During the sitting and standing activities, the relative muscular movements do not work in order to move a part of the body but to maintain the equilibrium among all the sub-systems. In fact, the postural equilibrium aims to maintain the body in a specific position against the external forces, as gravity or perturbing factors. During standing and sitting activities the *Gastrocnemius*, Hamstrings, *Gluteus Maximus*, *Erector Spinae* and *Rectus Abdominalis* present dissimilar behaviors (Ashford & De Souza, 2000). In particular, sEMG values of the *Erector Spinae* are commonly lower in the sitting activity than in the standing activity (Cram et al., 1998).

Many muscles are involved in the walking activity. In particular, the most active muscles are the distal ones, as the *Soleus*, *Tibialis Anterior* and *Gastrocnemius*, while those proximal are less active (Winter & Yack, 1987). During the walking, also the Hamstrings, *Gluteus Maximus* and *Erector Spinae* are sites of interest (Cram et al., 1998). Many muscles have a similar profile while running and walking. The only exception is that the muscles of the calf group, as *Soleus*, *Gastrocnemius Medialis*, *Gastrocnemius Lateralis* and *Peroneus Longus* activate earlier during the running activity than in the walking activity (Gazendam & Hof, 2007).

In the cycling activity, the *Gluteus Maximus* and *Biceps Femoris* are important for the hip extension. Also the knee extension and flexion are important for the production of force. For the knee extension the *Rectus Femoris*, the *Vastus Lateralis* and *Vastus Medialis* play an important role; whereas the *Semimembranous*, *Biceps Femoris* and *Gastrocnemius* play an important role for the knee flexion (Burke, 2002).

The number of muscles to be sensed should be limited in order to minimize the number of channels; that means making the system as less cumbersome as possible. Finally, considering the previous analysis combined with usability reasons and social acceptance, the following muscles have been chosen: *Tibialis Anterior*, the *Gastrocnemius*, the *Vastus Lateralis* and the *Erector Spinae*. The selected muscles are labeled in Figure 5.2.

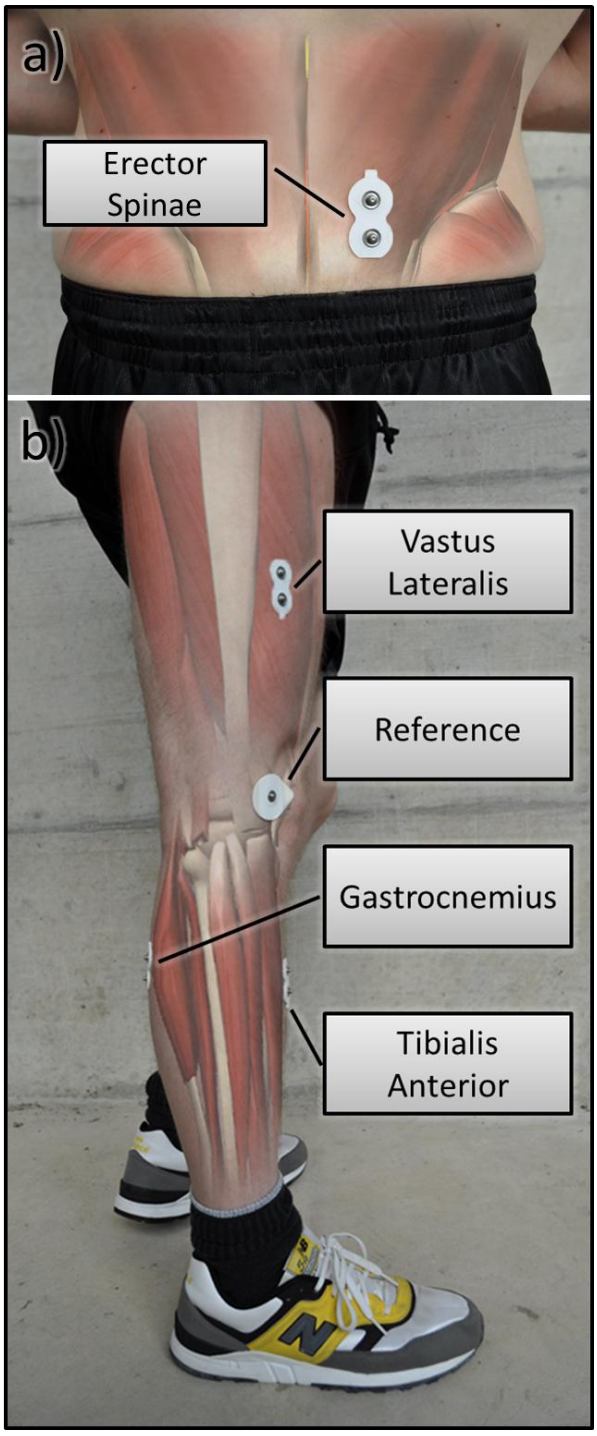


**Figure 5.2 Visualization of the selected muscles a) on the back and b) on the leg, and the relative sensors (white electrodes) placement.**

In order to achieve the correct electrodes placement, it has been taken as reference the instructions reported in (Cram et al., 1998), and by the Surface Electromyography for the Non-Invasive Assessment of Muscles (SENIAM) project (Hermens et al., 1999). For the sensor placement on the *Erector Spinae*, it is

necessary to look for the iliac crest in order to determine the L-3 vertebra. Then, the electrodes must be placed parallel to the spine at the level of the L-3 vertebra and approximately 2 cm from the spine. For the *Tibialis Anterior* sensing, the electrodes were placed over the largest muscles mass situated approximately one-quarter to one-third the distance between the knee and the ankle. Further details concerning the correct sensors placement for the *Erector Spinae* and the *Tibialis Anterior* may be found in (Cram et al., 1998). The *Gastrocnemius* EMG signal can be sensed placing the electrodes at one-third of the line between the head of the fibula and the heel. The *Vastus Lateralis* electrodes need to be placed at two-thirds on the line from the *Anterior Spina Iliaca Superior* to the lateral side of the patella. Further details concerning the correct sensors placement for the *Gastrocnemius* and the *Vastus Lateralis* may be found in [20]. The reference electrode has been placed over an inactive tissue (tendons or bony parts) as suggested in (Hermens et al., 2000). All the electrodes have been placed on the user's dominant leg.

It has been chosen to develop a prototype following the wearable paradigm. This prototype was composed of two different parts: the acquisition and the processing modules (as depicted in Figure 5.3). This prototype has been conceived to be full wearable, with the computing tasks delegated to a wearable computer. The two modules communicated via Bluetooth constituting a Personal Area Network. During the experiment session, it has been used a laptop.

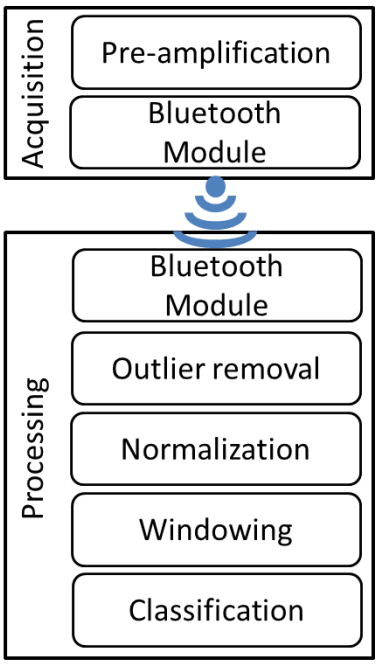


**Figure 5.3 System architecture.**

The wireless MQ16 device produced by Marq Medical has been used to detect the EMG signal. The chosen sensors were wet Ag/AgCl electrodes that provide fast signal response with low impedance. A custom driver has been developed to sample the signal at 1024 Hz and with 16-bit resolution. The driver also managed the wireless communication via Bluetooth with the processing unit. The raw data retrieved from the device are elaborated according to the stages listed below.

*Outlier removal*. The signal has been cleaned by identifying and removing the outliers from the signal.

*Normalization*. The normalization technique was such that it removed any amplitude differences between each muscle and subject. The data were normalized to give the same weight to the signals, independently from the muscle strength (Winter & Yack, 1987). Figure 5.4 reports an example of the elaborated sEMG signals for every activity.

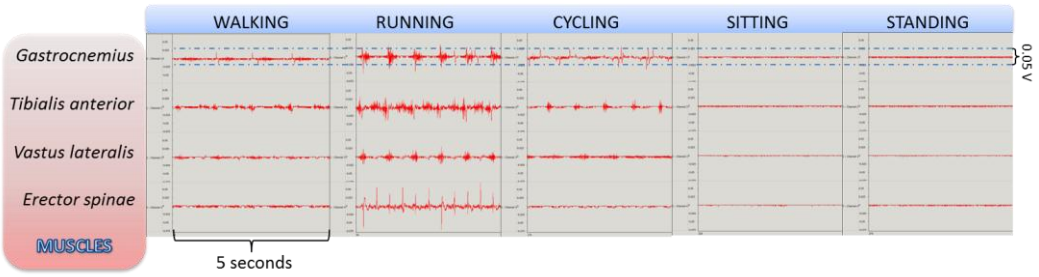


**Figure 5.4 EMG signals of every sensed muscle for the five activities.**

*Windowing*. The system was configured to work with 4 channels; each of them was associated to a specific muscle. The signal parts relative to the activities were extracted from every channel. The signal was segmented in different windows for every activity: the windows were 5 seconds long with an overlapping of 3.5 seconds. Those values in the pre-test phase showed the best results as compromise between accuracy and computational complexity.

*Classification*. The classification was based on the HMMs. In particular, in this prototype the continuous density HMMs were adopted as classifier and the observation data probability was modeled as a Gaussian distribution. During the training phase, it has been applied the Baum-Welch algorithm based on the forward-backward algorithm (Welch, 2003). An ergodic HMM with two hidden states was created for every activity.

## 5.3 Test

Eight subjects took part to this test (two females) with variable muscle mass and age distributed between 23 and 31 years. In Table 5.1, the age, the gender, the body weight, the height and are reported for each subject. The BMI was calculated as weight in kilograms divided by height in meters squared.

**Table 5.1 Test subjects' physical characteristics**

| | SUBJECTS | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | ***S1*** | ***S2*** | ***S3*** | ***S4*** | ***S5*** | ***S6*** | ***S7*** | ***S8*** |
| ***Age (in years)*** | 25 | 26 | 25 | 31 | 25 | 27 | 31 | 23 |
| ***Gender*** | F | M | M | M | F | M | M | M |
| ***Weight (in kg)*** | 56 | 85 | 91 | 69 | 54 | 74 | 80 | 53 |
| ***Height (in cm)*** | 167 | 176 | 187 | 175 | 164 | 179 | 175 | 155 |
| ***BMI (in kg/m$^2$)*** | 20.08 | 27.44 | 26.02 | 22.53 | 20.08 | 23.09 | 26.12 | 22.06 |

The test protocol began with the electrodes placement. The sensors were placed by three different people, without particular knowledge on physiology or medicine, taking as reference the results of our previous analysis.

The test took part in a gym. The participants were asked to perform the following activities: "walking", "running", "cycling", "sitting" and "standing". "Walking" and "running" were performed on a treadmill. The treadmill was configured with a constant speed of 3 km/h for the first task and 6 km/h for the second one, according to (van Ingen, 1979); the subjects used an exercise bicycle to perform the "cycling" activity. Participants used a normal chair, for "sitting", adopting a comfortable position with both feet on the ground and the rested on the chair back.

The entire set of activities was repeated two times. Each activity lasted thirty seconds with thirty seconds of interval between two subsequent tasks in order to give the time to the participants to prepare themselves for the following activity. The whole test lasted about thirty minutes per person.

During this test, 17 different samples per activity per person in one session were recorded; that means 34 samples per activity per person in the whole experiment. The total number of samples per activity is 272.

## 5.4 Analysis and Results

Two different kinds of analysis were conducted: the impersonal analysis and the subjective analysis. The first analysis had the goal of evaluating the generality of our approach. The data coming from all the subjects were mixed in order to build the training set and the test set for the classification. The second one was subject- oriented: the HMMs classifiers have been trained with data of a certain subject; then, the resulting trained classifiers were tested on other data coming from the same subject.

The impersonal analysis consisted of two parts: a generic k-fold cross-validation and a Leave One Subject Out (LOSO) analysis. In the first part, all the participants' data were shuffled and balanced among the classes and among the users. The k = 10 was chosen because it is a value commonly used in literature (Refaeilzadeh et al., 2009). The aim of the impersonal analysis was to validate the model over a group of data coming from different subjects. During the LOSO analysis, each of the eight users was used once for testing, whereas the remaining users were used for training. In this strategic way, the users' data independence can be evaluated.

The subjective analysis consisted in an elaboration specialized on a single user. In this case, it was investigated how a subject-oriented approach would improve the global performances obtained in the impersonal analysis. For every participant, the 10-fold cross-validation was performed. With the exception of the dataset selected for test and training, the data processing is the same used for the cross-validation of the impersonal approach.

Table 2 shows the results of the cross-validation of the impersonal analysis. It reports a mean accuracy of 96.8%. This result validated the robustness of the classifier, in spite of the physiological differences among the participants. The related confusion matrix showed excellent performances of the classifier above all the five classes, with a maximal error of 13.6% between "running" and "walking". In the other classes, the error was always lower than 1.5%.

**Table 5.2 Confusion matrix of the cross-validation with impersonal analysis (k=10; 272 samples per activity).**

| | | Recognized activity | | | | |
|---|---|---|---|---|---|---|
| | | *Walking* | *Running* | *Cycling* | *Sitting* | *Standing* |
| Labeled activity | *Walking* | 272 | 0 | 0 | 0 | 0 |
| | *Running* | 37 | 235 | 0 | 0 | 0 |
| | *Cycling* | 0 | 0 | 270 | 2 | 0 |
| | *Sitting* | 0 | 0 | 0 | 272 | 0 |
| | *Standing* | 0 | 0 | 0 | 4 | 268 |

| | |
|---|---|
| *Average (%)* | *96.84* |
| *Standard deviation* | *5.87* |

The LOSO study of the impersonal analysis (Table 5.3) confirmed the previous results. The accuracy mean of 91.8% highlighted the good system capability of adapting to unseen subjects. The confusion matrix confirmed that the most challenging task was to distinguish “running” from “walking”, with the 28.7% of errors. The LOSO study showed also an increasing number of errors between “standing” and “sitting” (12.5%).

**Table 5.3 Confusion matrix of the LOSO (subjects number =8; 272 samples per activity).**

| | | Recognized activity | | | | |
|---|---|---|---|---|---|---|
| | | *Walking* | *Running* | *Cycling* | *Sitting* | *Standing* |
| Labeled Activity | *Walking* | 272 | 0 | 0 | 0 | 0 |
| | *Running* | 78 | 194 | 0 | 0 | 0 |
| | *Cycling* | 0 | 0 | 272 | 0 | 0 |
| | *Sitting* | 0 | 0 | 0 | 272 | 0 |
| | *Standing* | 0 | 0 | 0 | 34 | 238 |

| | |
|---|---|
| *Average (%)* | *91.76* |
| *Standard deviation* | *12.64* |

The difference between the cross-validation and the LOSO results was noteworthy. It showed the behavior of the system in the classification of new

and unseen subjects' data. This difference was also influenced by small differences due to a slightly imprecise sensors placement. These observations have been confirmed by the accuracy rates reported in Table 5.4, which shows the results of the subjective analysis. In this case, the training set and the test set came from the same subject avoiding the issues introduced by the difference among participants. The cross-validation showed an increased average accuracy of 99.4%. The system achieved the 100% of accuracy on more than half of the subjects. In this case, the errors were shared among the different activities without significant variations.

**Table 5.4 Subjective analysis.**

| | SUBJECTS | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | ***S1*** | ***S2*** | ***S3*** | ***S4*** | ***S5*** | ***S6*** | ***S7*** | ***S8*** |
| ***Accuracy (%)*** | 100 | 100 | 98.33 | 99 | 100 | 100 | 97.67 | 100 |
| ***Average (%)*** | ***99.38*** | | | | | | | |
| ***Standard deviation*** | ***1.84*** | | | | | | | |

As expected, the accuracy results showed that the subjective approach allows achieving better results than with the impersonal one (see Figure 5.5). This means that the physiological differences from person to person and also the slight differences in the sensors placement did not allow the training of a single classifier for several users with the same effectiveness of a classifier adapted to a single user. This is a typical problem in physiological signals analysis and it must be seriously taken into account for an out-of-the-lab application. In this experiment, the prototype needed one minute of training per activity in order to generate a dedicated classifier for a particular subject. This slim amount of time spent for the training is justified by the conspicuous improvement of performances. For this reason, the best approach could be providing an impersonal classifier for the first utilizations but providing the possibility of a subsequent specialization to the user peculiarity.

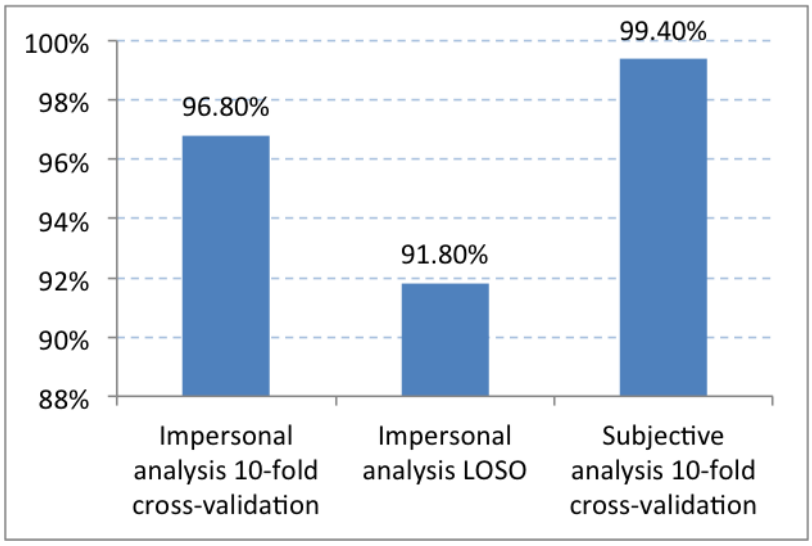


**Figure 5.5 Activity recognition accuracy rates for the impersonal analysis with 10-fold cross-validation and LOSO, and subjective analysis with 10-fold cross-validation.**

## 5.5 Optimization

In order to estimate the relationship between the used channels and the activity recognition accuracy, the number of channels was progressively reduced trying all the possible combinations; then, the achieved accuracy was compared with the values reached during the reference experiment that has been realized using the four channels configuration. In this case, *configuration* indicates the ensemble of specific channels, where a channel corresponds to a single sensed muscle. In particular, channel 1 corresponds to the *Gastrocnemius*, channel 2 to the *Tibialis Anterior*, channel 3 to the *Vastus Lateralis* and channel 4 to the *Erector Spinae*. A *category* comprises all the permutations for a given number of channels. As for the reference case, the data analysis has been effectuated in two conditions: impersonal and subjective.

In the impersonal analysis, the accuracies and the confusion matrixes relative to the configurations with one, two and three channels were calculated. A total of fourteen different configurations have been tested. The results are summarized in Table 5.5.

**Table 5.5 Accuracy rates in the impersonal analysis.**

| Impersonal Analysis – Accuracy Rates | | |
|---|---|---|
| ***Category (# Channels)*** | ***Configuration*** | ***Results*** |
| 4 | 1-2-3-4 [a] | 96.8% |
| 3 | 1-2-3 | 96.5% |
| | 1-2-4 | 94.7% |
| | 1-3-4 | 95.7% |
| | 2-3-4 | 90.9% |
| 2 | 1-2 | 83.3% |
| | 1-3 | 84.1% |
| | 1-4 | 93.3% |
| | 2-3 | 90.5% |
| | 2-4 | 83.5% |
| | 3-4 | 74.7% |
| 1 | 1 | 75.0% |
| | 2 | 62.1% |
| | 3 | 64.9% |
| | 4 | 77.7% |

The reference case has been performed using all the four channels; this test provided an accuracy rate of 96.8%. For the three channels category, the best accuracy was attained without the channel 4 (96.5%). Comparing the accuracy rates of the configurations present in the three channels category with the reference case, it is possible to observe that the difference among them is not large. In particular, the configuration using channels 1-2-3 has a difference of 0.3% from the reference case. This means that the information provided from the four channels is redundant for these five activities. Among the two channels

system tests, the best accuracy was obtained using the channels 1 and 4 achieving a recognition accuracy of 93.3%.

Table 5.6 shows the confusion matrixes using one singular channel for the activity recognition. From these results it is possible to deduce that, first, this prototype using the information provided from channel 1 was able to recognize standing, cycling and running with high accuracy (over 88.2%); it achieved also good performances for walking recognition (65.8%), but it misclassified the 71.7% of sitting samples. Moreover, this prototype using the information provided from channel 2 was able to recognize walking, sitting and standing activities with very high accuracy (over 97.4%); in this case, the running and cycling activities achieved low accuracy rates. Furthermore, this prototype using the information provided from channel 3 was able to recognize running, cycling and sitting with high accuracy rates (over 80.5%); it misclassified about 50% of walking data and it provided a very low accuracy for the standing recognition. Finally, the channel 4 provided significant information for four activities (over 84.9%) with the exception of cycling.

Comparing the results in the one channel category, it is possible to notice that channel 4 provided the best accuracy rate. This means that even if the channel 4 signal is the feeblest in terms of voltage, it allows the system to reach a remarkable accuracy of 77.7%. Another interesting result, in terms of accuracy, was obtained using only the channel 1 (75%). The accuracy rates provided from the two channels category configurations are dependent upon the accuracy rates provided by the one channel category configurations. In particular, in the two channels category the configuration that provided the best accuracy result was composed of channel 1 and channel 4. These two channels separately achieved the best results among the configurations in the one channel category.

**Table 5.6 Confusion matrixes for the configurations of the one channel category.**

| **Impersonal Analysis – Confusion Matrixes** | | | | | |
|---|---|---|---|---|---|
| ***Channel 1*** | ***Walking*** | ***Running*** | ***Cycling*** | ***Sitting*** | ***Standing*** |
| **Walking** | 179 | 93 | 0 | 0 | 0 |
| **Running** | 32 | 240 | 0 | 0 | 2 |
| **Cycling** | 18 | 0 | 254 | 0 | 9 |
| **Sitting** | 0 | 0 | 0 | 77 | 195 |
| **Standing** | 0 | 0 | 0 | 2 | 270 |
| **Average** | 75.0% | | | | |
| ***Channel 2*** | ***Walking*** | ***Running*** | ***Cycling*** | ***Sitting*** | ***Standing*** |
| **Walking** | 266 | 6 | 0 | 0 | 0 |
| **Running** | 250 | 17 | 5 | 0 | 0 |
| **Cycling** | 100 | 56 | 24 | 0 | 92 |
| **Sitting** | 0 | 0 | 0 | 272 | 0 |
| **Standing** | 7 | 0 | 0 | 0 | 265 |
| **Average** | 62.1% | | | | |
| ***Channel 3*** | ***Walking*** | ***Running*** | ***Cycling*** | ***Sitting*** | ***Standing*** |
| **Walking** | 141 | 106 | 0 | 3 | 20 |
| **Running** | 53 | 219 | 0 | 0 | 0 |
| **Cycling** | 0 | 0 | 272 | 0 | 0 |
| **Sitting** | 19 | 0 | 0 | 226 | 27 |
| **Standing** | 58 | 0 | 0 | 188 | 26 |
| **Average** | 64.9% | | | | |
| ***Channel 4*** | ***Walking*** | ***Running*** | ***Cycling*** | ***Sitting*** | ***Standing*** |
| **Walking** | 272 | 0 | 0 | 0 | 0 |
| **Running** | 0 | 270 | 2 | 0 | 0 |
| **Cycling** | 0 | 0 | 12 | 57 | 203 |
| **Sitting** | 0 | 0 | 0 | 231 | 41 |

| | | | | | |
|---|---|---|---|---|---|
| **Standing** | 0 | 0 | 0 | 0 | 272 |
| **Average** | 77.7% | | | | |

The subjective analysis was conducted only for the best configurations obtained during the impersonal analysis in order to evaluate the data relative to single subjects. For each subject, the data were distributed in a training set and a test set, and the 10-fold cross-validation was adopted for the analysis. The results of the comparative tests are shown in Table 5.7. As expected, there has been an improvement of the accuracy in every instance; this is due to fact that the training and test sets came from the same person.

**Table 5.7 Accuracy rates in the subjective analysis compared to the impersonal analysis for the selected configurations.**

| *Category (# Channels)* | *Configuration* | *Results: Impersonal* | *Results: Subjective* |
|---|---|---|---|
| 4 | 1-2-3-4 [a] | 96.8% | 99.4% |
| 3 | 1-2-3 | 96.5% | 99.5% |
| 2 | 1-4 | 93.3% | 97.2% |
| 1 | 1 | 75.0% | 81.1% |
| | 4 | 77.7% | 87.0% |

In the case of one channel category, the accuracy is largely higher than in the impersonal 10-fold cross-validation. The confusion matrixes of channels 1 and 4 are reported in Table 5.8.

In the subjective analysis, the prototype set to use the data provided from channel 1 achieved an average accuracy rate of 81.1%. In particular, it was able to recognize running and cycling with high accuracy; it achieved also good performances for the other three activities. The prototype using channel 4

achieved an accuracy rate of 87%. In the two channels category, the configuration always using the channels 1 and 4 provided a very high accuracy: 97.2%.

In the subjective analysis, it was possible to notice the same effect previously highlighted in the impersonal analysis. The accuracy rate obtained in the reference case is almost equal to the accuracy obtained in the configuration using the channels 1-2-3. This observation implies that the information provided from the channel 4 is redundant if put in relation with the combination of the other three channels. In fact, the single channel 4 data contain the information necessary for the recognition of all the five activities: the system using only this channel achieved a considerable accuracy rate.

**Table 5.8 Confusion matrixes for the selected configurations of the one channel category.**

| **Subjective Analysis – Confusion Matrixes** | | | | | |
|---|---|---|---|---|---|
| ***Channel 1*** | ***Walking*** | ***Running*** | ***Cycling*** | ***Sitting*** | ***Standing*** |
| **Walking** | 185 | 87 | 0 | 0 | 0 |
| **Running** | 1 | 271 | 0 | 0 | 0 |
| **Cycling** | 13 | 0 | 259 | 0 | 0 |
| **Sitting** | 17 | 1 | 0 | 194 | 60 |
| **Standing** | 37 | 0 | 0 | 42 | 193 |
| **Average** | 81.1% | | | | |
| ***Channel 4*** | ***Walking*** | ***Running*** | ***Cycling*** | ***Sitting*** | ***Standing*** |
| **Walking** | 232 | 40 | 0 | 0 | 0 |
| **Running** | 36 | 236 | 0 | 0 | 0 |
| **Cycling** | 0 | 0 | 272 | 0 | 0 |
| **Sitting** | 0 | 0 | 0 | 227 | 45 |
| **Standing** | 0 | 0 | 0 | 53 | 219 |
| **Average** | 87.0% | | | | |

Using a reduced number of channels allows applying fewer sensors on the body: this means improving the whole system in terms of usability and computational cost. The computational cost performance has been assessed in terms of CPU time among the different categories. The results are shown in Figure 5.6. The experiment was performed on a desktop computer with an Intel Core i7 860 CPU and 4 GB RAM running Windows 7 Enterprise edition (64-bit). As expected, reducing the number of channels means decreasing the computational complexity. In the reference case, the classification of a 5 seconds window required an average time of 615 ms. In the three channels category, the computational cost has been estimated of 467 ms. For the two channels category, the average computational time was 331 ms, whereas in the one channel category it decreases to 181 ms.

Figure 5.6 shows the accuracy rates for both impersonal and subjective analyses in relation with the computational cost for every category. In particular, it is important to highlight that, in terms of accuracy rate, the reference case and the three channels category provided very similar performances; however, the three channels category allowed reducing the computational cost of 24.1% compared to the reference case.

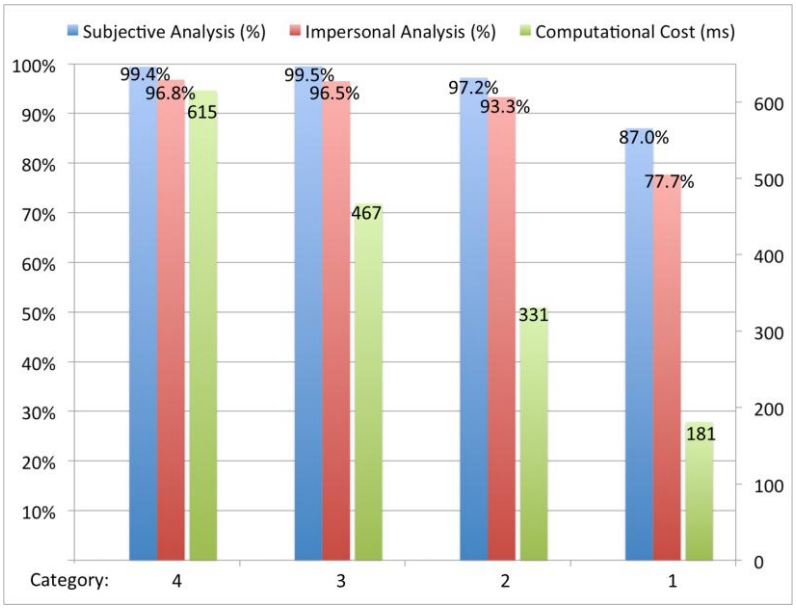


**Figure 5.6 Accuracy rates of the subjective and impersonal analyses for the four categories with the relative computational costs.**

## 5.6 Summary

This chapter presented a further development of context awareness for ubiquitous computing interfaces. As already stated, the user activity is very relevant in the frame of "understanding" the context of a specific interaction. The pervasive technology and, in particular, the wearable paradigm allows sensing human activity at a new level: muscular activity. The application of sEMG-based technology on the lower limbs for activity recognition is a novelty in HCI. The work presented in this chapter shows how the information provided by sEMG signals can be elaborated in order to extrapolate the user physical activity, which is strictly related to the overall activity. In fact, the approach presented in this thesis allows recognizing whether the user is walking, running, cycling, standing and sitting. The wearable approach allows the system to monitor the user activity everywhere; this information can be used to provide always the best service at the opportune moment. The EMG signal analysis does not consist only of a study of the electrical signal but it implies also a preliminary study for the

choice of the muscles to sense. In this chapter, the study of the human anatomy concerning the selection of the body areas of interest is presented in order also to allow other researchers to reproduce the tests. The signal analysis is mainly based on machine learning techniques, i.e., HMM classifier. The results show that this approach is valuable for the user physical activity recognition; moreover, it has been shown that the recognition accuracy can be improved starting from a general system that has never been used by the user, to a specialized personal system that is trained on the user. This observation acquires value when relating this process to a commercial system: an off-the-shelf device should provide a minimum accuracy rate that should adequate to the specific task, in this case activity recognition. However, this system can specialize on the user during the use: if the user wears the system, the latter can modify the classifier in order to adapt the model to the peculiar characteristics of the user, improving the recognition accuracy. This difference in terms of accuracy performances can be noticed reading the difference between the LOSO and subjective analyses showed in Figure 5.5.

The choice of using EMG-based technology provides two main advantages: the first is that EMG signals carry a lot information also about the user's physical health status, which can be used in the future in order to allows smarter and smarter services; the second advantage consists of the possibility of integrating this kind of sensors in clothes allowing the development of real unobtrusive wearable systems (Linz et al., 2007). The possibility of leveraging EMG-based activity recognition integrated in clothes is important in the ubiquitous computing era, in particular, with reference to the pervasive computing paradigm. In fact, it can be imagined that users wearing smart clothes able to recognize the activity could transmit the information to the smart environment in order to improve the interaction experience. This synergy could allow to a user to experience a seamless interaction while moving through different smart environments, such as a living room, an office or a vehicle. This concept will be presented and discussed in the next chapter.

# Chapter 6: System Status: the Synergistic Paradigm

## 6.1 Introduction

Context information comprehends also the information concerning the system status. In fact, also the typology of distribution and the availability of sensors influence the interaction. As presented in the subsection 2.4.1 of the literature review, the gestural interaction systems can adopt three paradigms in the ubiquitous computing era: environmental, wearable, and pervasive computing (see Figure 6.1). The environmental computing paradigm uses distributed sensors in the interaction space to detect and react to the inhabitant's movements, gestures, and activities. The wearable computing paradigm uses sensors worn by the person for detection and sensing. The pervasive computing paradigm mixes the environmental and wearable computing paradigms. The environmental and wearable computing paradigms have different and complementary advantages and drawbacks. In particular, (Rhodes et al., 1999) evaluated the advantages and disadvantages of these two paradigms on seven specific features: privacy, personalization, localized information, localized control, and resource management. While the wearable computing paradigm presents problems the last three features, the environmental paradigm is particularly ill suited for the privacy and personalization. (Rhodes et al., 1999) stated that mixing wearable and environmental computing allows having the advantages of both; as proof-of-concept, they presented a peer-to-peer network of wearable and environmental systems, which mixes the complementary advantages coming from both paradigms. To be more precise, this architecture

aims at bringing together privacy and personalization provided by the wearable paradigm with localized information, localized control and resource management characterizing the environmental paradigm. The work presented in (Carrino et al., 2011) wanted to extend the previous concept of mixing together the benefits driven from the adoption of these paradigms introducing also the consistency feature. The architecture proposed by the authors focuses on gesture recognition and deals with the features and classifier results fusion. In fact, the fusion allows providing better recognition accuracy.

In this thesis, this concept is further extended and formalized in a novel paradigm of ubiquitous computing that is defined synergistic paradigm. The synergistic paradigm is composed of a wearable subsystem and an environmental subsystem. These two subsystems are dynamically managed in relation with their contextual availability in order to provide seamless gesture recognition. The wearable and environmental subsystems can function independently but if combined they can exploit a fusion engine, which allows increasing the gesture recognition accuracy. Implementing the opportunistic paradigm allows developing a system combining the advantages coming from both the wearable and environmental subsystems augmenting the interaction possibilities. Moreover, it grants the gesture recognition accuracy being always no lower than the best accuracy obtained with the single subsystem. Providing a system that is always available and able to contextually increase the gesture recognition accuracy means also creating a better user experience.

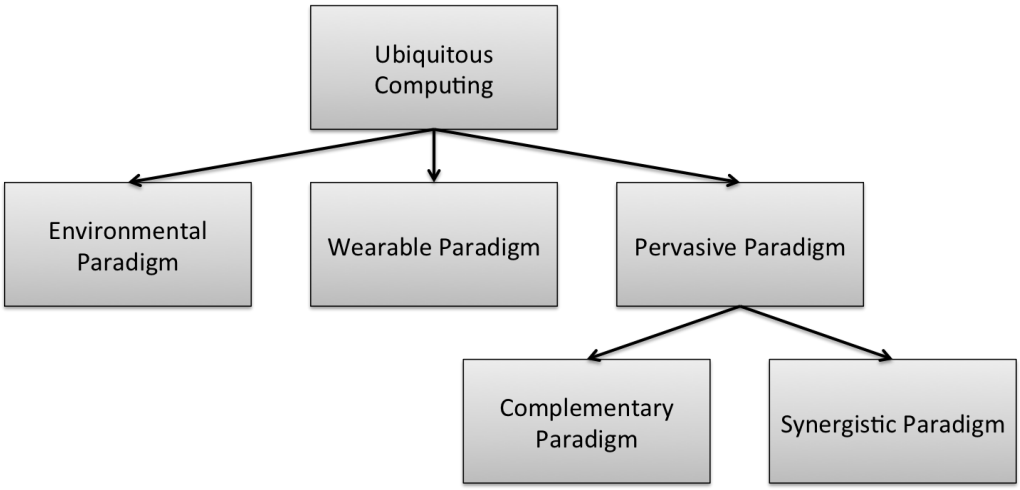


**Figure 6.1 This diagram shows the different paradigms for the design and development of gestural interfaces in the ubiquitous computing era.**

## 6.2 The paradigms

The pervasive computing paradigm can be divided into two types: the complementary and synergistic. The complementary type requires that the environmental system and the wearable system are always available; in fact, only the simultaneous presence of both systems permits its functioning. Usually, the complementary type involves the adoption of early fusion approach in order to enhance the recognition accuracy. This approach is a general class of methodologies working with the information before a classifier elaborates it. The early fusion can be further split in two sub-levels: data level and features level, as shown in Figure 6.2 and Figure 6.3.

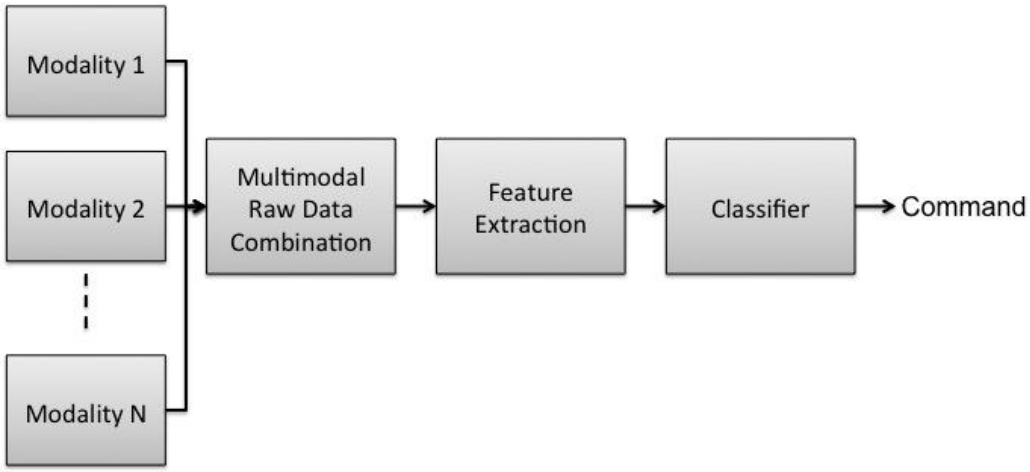


**Figure 6.2 This diagram describes the early fusion approach at data level.**

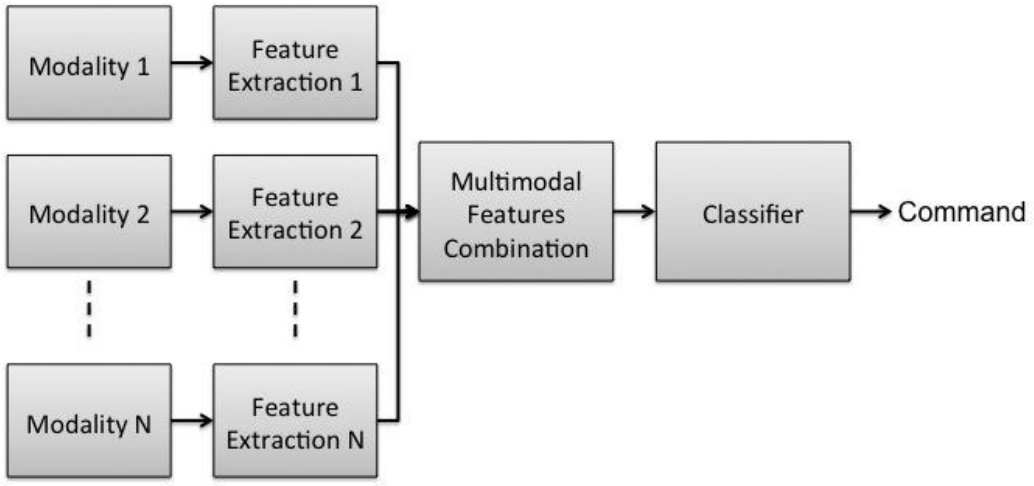


**Figure 6.3 This diagram represents the early fusion approach at feature level.**

The complementary type allows merging the information at early stages, which involves a reduced redundancy of computation improving the efficiency.

The synergistic type, on the contrary, is more dynamic and allows the wearable and environmental systems to function independently as well as to merge their information when both are available. This type of pervasive paradigm boasts not only this opportunistic approach that allows enhancing the recognition accuracy but it allows the dynamic fusion of different sensors that can augment the interaction possibilities. In fact, if one of the wearable and environmental systems is not available because of any reason, the general system will keep

functioning in order to allow a seamless and continuous interaction, even if it could imply a loss of performances in terms of recognition accuracy.

Here the definition is reported: an opportunistic multi-component system is synergistic if all the components contribute to the overall system performances, but the lack of some of them does not compromise the proper functioning of the whole system. A system with multiple components should perform better than its individual components. Performances are measured in terms of interaction, usability or accuracy.

The synergistic type of pervasive paradigm usually implies the adoption of a late (or decision level) fusion. Whereas the early fusion approaches merges modalities in the features space, the late fusion (called also decision level fusion) fuses modalities in the semantic space. In the late fusion strategy, the classifiers are included in the fusion scheme (see Figure 6.4).

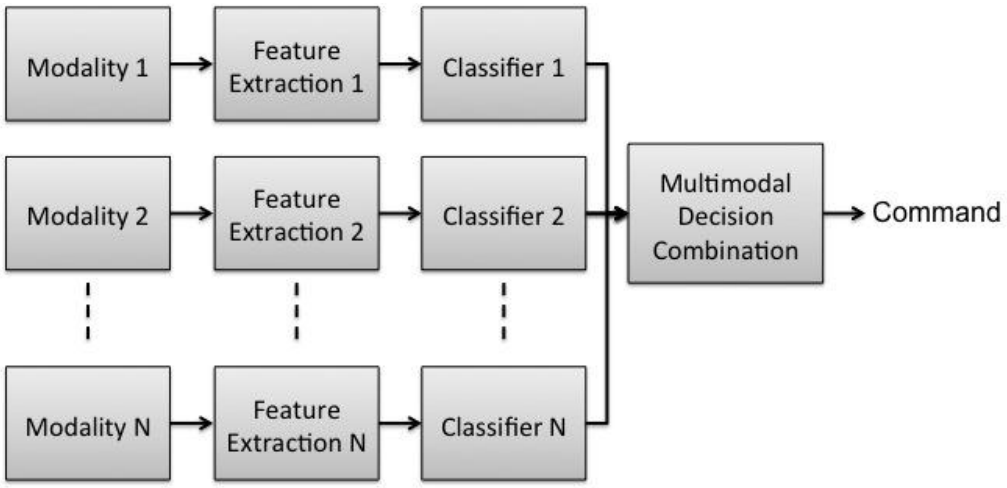


**Figure 6.4 This diagram depicts the late fusion approach.**

In literature, it is possible to identify three fusion sub-levels according to the kind of output information provided by the different classifiers:

- Abstract-level (Faltemier et al., 2006): whereas the classifiers give as output just the label associated to the nearest class.
- Rank level (Ben Soltana et al., 2010): whereas the classifiers output a rank list of class label in order of similarity.

- Measurement-level (or score match level) (Cook et al., 2007): whereas the classifiers produce similarity scores.

The two main reasons for combining classifiers are efficiency and accuracy. In order to increase efficiency, multistage combination rules can be adopted; in this case, a simple classifier using a small set of cheap features in combination with a reject option classifies the signals. There are several advantages in late fusion approaches. Firstly, it can avoid strict time limitations (Jaimes & Sebe, 2007); secondly, the training requirements are smaller and the development process is simpler using a late integration method (Turk, 2005). However, the learning phase requires training more classifiers, even if with inferior complexity. (Ben Soltana et al., 2010) showed that a disadvantage of the late fusion approach is the possible loss of correlation in mixed features space and in the case the individual experts (or classifiers) are correlated, it may not be the best scheme to follow. Moreover, the complementary type allows reducing the computational redundancy allowing a better exploitation of the hardware resources.

The dynamic adaptability of the system to its own status introduces a new element in the context awareness of the system. This new element of a “self aware” system allows improving the gesture recognition accuracy and leveraging more features than the wearable, environmental and complementary paradigms.

## 6.3 Proof-of-Concept

A prototype has been developed in order to conduct some experiments to validate this system design approach for the pervasive computing paradigm. The cockpit of a car has been selected as smart environment for this scenario. In fat, every year more and more people spend a considerable part of their life in cars. Re-cent statistics demonstrated that the average Swiss resident drove 23.8 km per day in 2010 (Swiss Federation, 2011); in the U.K., the average motorist spent 10 hours per week in the car (The Telegraph, 2011). For this reason, carmakers are trying to make this “in-vehicle life” more enjoying, by equipping the car with various In-Vehicle Infotainment Systems (IVISs). All these systems need to be

controlled by the car inhabitants and the common approach is to position most of these controls in the central dashboard, in order to make them accessible also to the passenger. Typical approaches make use of knobs and buttons, but over the years many carmakers have replaced these primordial systems with touchscreens, or advanced haptic controls like the BMW i-Drive (Niedermaier et al., 2009). When control systems are placed in the central dashboard, the driver has to leave one hand from the steering wheel and the eye gaze from the road. According to (Bach et al., 2009), most cases of general withdrawal of attention are caused by the loss of visual perception, often because of eyes-off-the-road distraction. Natural interaction addresses this issue enhancing the user experience while reducing the cognitive load induced by secondary tasks. In particular, gestural interaction represents "promising means to cover the full range of a driver's operational needs while minimizing the cognitive and visual workload", as stated in (Riener, 2012).

### 6.3.1 Interface design

The design of the system began with the definition of the gesture taxonomy. The first point to define was the choice of the interaction space for the gesture performance. The study of the scientific literature and a brief observation of the driver behavior in real-case scenarios identified the steering wheel as the best interface for the in-vehicle gestural interaction. In fact, while free-hand gestures could be troublesome for the driver, gestures performed on the steering wheel appear as a safer natural interaction approach. This conclusion has been reported in many different works: (e.g., Endres et al., 2011; Pfleging et al., 2011; Döring et al., 2011; Gonzalez et al., 2007; Murer et al., 2012). In fact, allowing the driver to perform gestures while holding the steering wheel helps keeping the eyes on the road while interacting.

Performing the gesture while grasping the steering wheel leads to the concept of tangible gesture interface. Indeed, tangible gesture interaction has been recently defined by Hoven and Mazalek as "the use of physical devices for facilitating, supporting, enhancing, or tracking gestures people make for digital interaction

purposes" (van den Hoven & Mazalek, 2011). Tangible gesture interaction still belongs to the broader field of tangible interaction, conjugating its most important property, i.e., physicality, and the communicative role of gestures. In this case, the physicality is brought by the steering wheel, which can be seen as a tangible interface not only for the driver's primary task, benefiting of the direct manipulation of the car behavior and of the haptic feedback from the road, but also for secondary tasks (Fishkin et al., 1999).

The gesture taxonomy has been chosen following the functional gesture procedure. First, the most important functions have been selected. The functions are for the IVIS control, in particular, for the media player: play, stop, next, and previous. The gesture design has been made with reference to (Wolf et al., 2011). In fact, Wolf et al. analyzed from an ergonomic point of view the possibility to use micro-gestures on the steering wheel to perform secondary tasks while driving. In particular, they identified some gestures that are particularly easy to perform while the driver holds the steering wheel. Following Wolf et al.'s analysis for the palm grasp, three gestures have been selected: tapping with the index and dragging fingers around the wheel (in both directions). A fourth gesture, squeezing, has been chosen even if it was not considered in the Wolf et al.'s analysis. In fact, this latter gesture requires minimal effort and cognitive load for the user. Moreover, several systems used squeezing as interaction modality with objects; (Harrison et al., 1998) showed some advantages of the squeeze gesture, for example the possibility to perform a squeeze without moving the hand from the object, which indeed is very useful while driving. In order to facilitate remembering the four chosen gestures, embodied metaphors have been used to associate the micro-gestures to their corresponding functions for the IVIS control (Bakker et al., 2012). The driver can start performing the tap gesture on the steering wheel in order to make some music: indeed, a single tap with the index is interpreted by the system as turning on the music. Dragging up and down the fingers on the steering wheel allows browsing up and down in the playlist. Squeeze is used to stop the music, which

intuitively binds closing the hand to closing the music player. Figure 6.5 depicts the selected gestures.

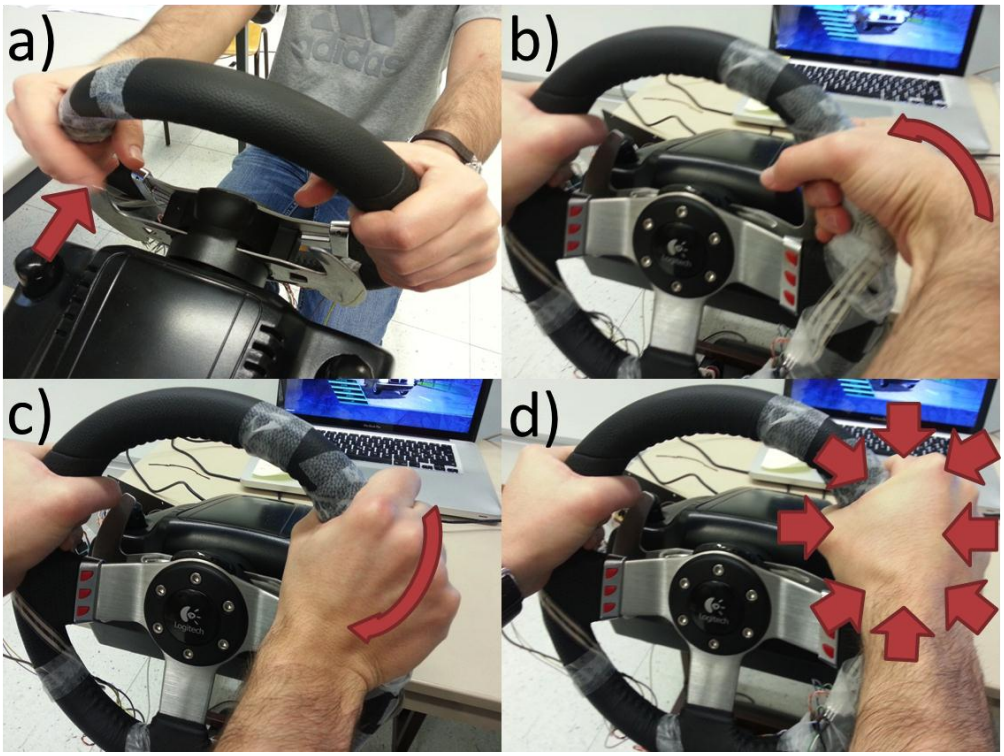


**Figure 6.5 Representation of the four gestures: a) tap, b) dragging up, c) dragging down, d) squeeze.**

The design of a gestural interface requires a proper feedback in order to acknowledge the user on the result of the given command. Tangible gesture interaction on the steering wheel involves doing an intensive use of haptic senses for the driver. Gestures are designed to be as intuitive as possible, without the need of visual attention on a graphic interface. Thus, it could be given as an opportune practice to convey the feedback to the user on the same haptic channel, using vibration motors or tactile displays. However, as stressed in (Bach et al., 2009), there is a risk of increased distraction if the secondary task competes on the perceptual resources required by the primary task. Indeed, haptic feedback coming from the road and perceived through the steering wheel has an important role in the driving task. As suggested in (Wickens, 1992), the perceptual resources needed for the secondary task can be distributed over

other senses. In this prototype, the auditory feedback generated by the media player has been chosen as feedback: tap and squeeze can be easily detected respectively by the presence or absence of the music. Dragging gestures can be identified with a change of the song: in the case of a playlist, the dragging up gesture is acknowledged by the music of a new song (next track), while the dragging down gesture corresponds to a song already listened (previous track). Obviously, the purpose of this application, i.e. listening to music, ensures that the auditory channel is not disturbed, thus the feedback is effective. In case of doubt, or for further information about the song, the user can still look at the screen on the central dashboard.

### 6.3.2. Implementing the environmental paradigm

In order to evaluate this gestural interface based on micro-gesture performed on the steering wheel, a prototype has been developed to conduct some usability tests. The sensing system implemented in the first prototype is based on five Tekscan FlexiForce sensors with a range of 0-1 lb. They are connected to an Arduino Duemilanove board that converts analogic signals to the digital domain and sends measured data to a PC for further elaboration through a wired serial connection. Data are acquired with a rate of 50 Hz. The pressure sensors have been integrated in a Logitech G27 Racing Wheel with four sensors dedicated to the right hand and one sensor for the left hand. The sensors placement on the steering wheel is depicted in Figure 6.6. Sensor 1 is placed to recognize the tap gesture with the index finger. In a relaxed position, the hand generally covers the three other sensors. The wrist flexion and the wrist extension performed for the dragging up and down gestures uncover respectively Sensors 3 and Sensor 4. Sensor 5 is used to segment gestures with the left hand in order to minimize false positives during the execution of the primary task: the driver squeezes the left hand while gestures are performed with the right hand. These five pressure sensors have been placed in the specific regions of the external ring in order to be compliant with the hands position suggested by the Swiss driving school manual.

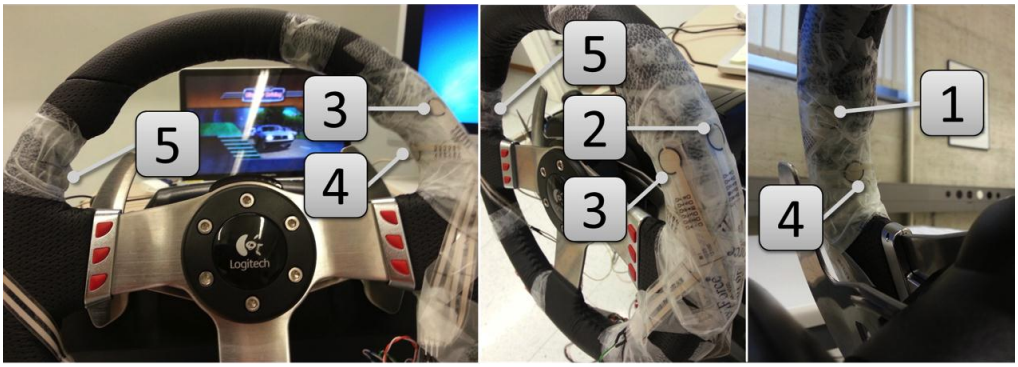


**Figure 6.6 The five FlexiForce sensors placement.**

The gesture recognition software was composed of three modules: an acquisition module, the segmentation module and the HMM classifier (Figure 4.7).

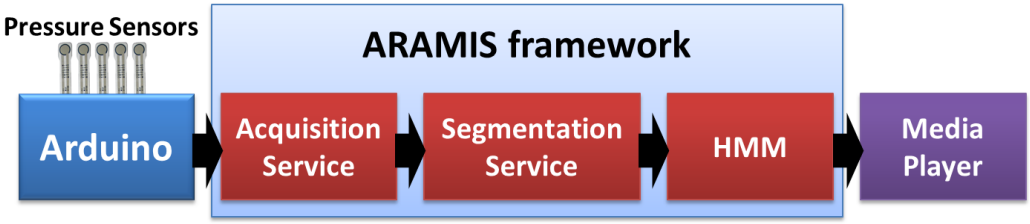


**Figure 6.7 Block diagram of the WheelSense system architecture.**

The HMM classifier was configured with 4 hidden states with forward topology and implementing the Baum-Welch algorithm to find the unknown parameters. The data supplied to the HMM classifier were modeled as temporal signals (as depicted in Figure 6.8).

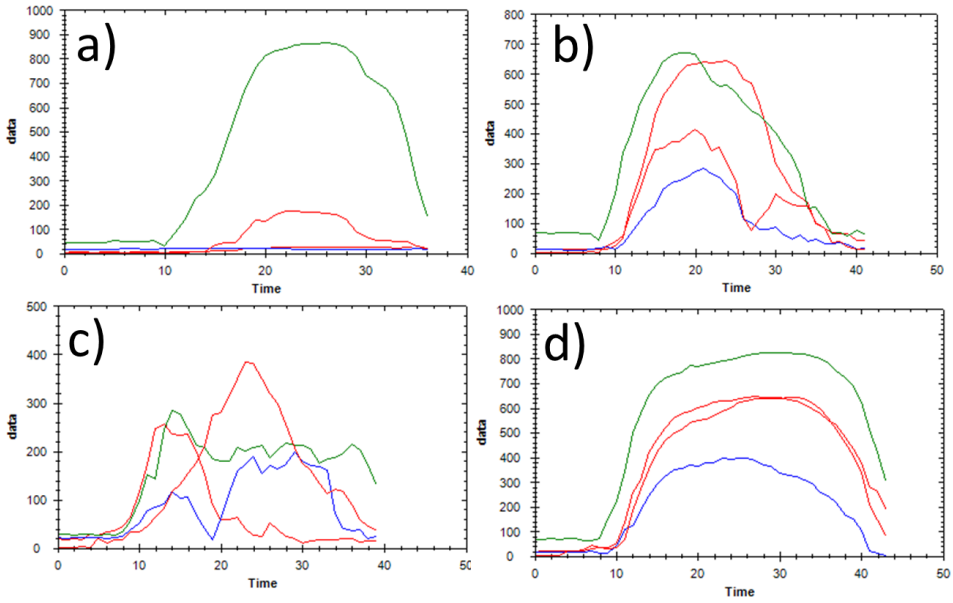


**Figure 6.8 Representation of the temporal signal associated to the four gestures: a) is tap, b) is dragging up, c) is dragging down and d) is squeeze.**

In order to assess the usability of this gestural interface, a usability test has been conducted. Eight users (six males and two females, aged 25 - 31) participated to this evaluation. The setup, depicted in Figure 6.9, is composed by a laptop that executes the recognition task and the City Car Driving Simulator version 1.2. The monitor on the right shows the results of the classification and the experiment supervisor used it.

**Figure 6.9 A testee interacting with the system during the evaluation.**

At the beginning of the test session, the users were asked to perform 40 times each gesture while the PC was recording for a total of 160 gestures. The order of the gesture to be performed was chosen randomly and the user was guided by a graphical interface. The user was requested to rest at half of the recording phase. The data recorded during this phase were used to train the HMM classifier on the specific user. Then, the users had to drive using the City Car Driving simulator and to interact with the IVIS through the gestural interface. The gestures to be performed were asked by the test supervisor; in particular, the total number of gestures that each user had to perform during the driving simulation was 40 (10 per type of gesture). Afterwards, the users had been asked to fill in the SUS questionnaire (Brooke, 1996). Three factors have been calculated from the SUS: the overall usability, perceived usability and the learnability. The overall usability (calculated following the standard procedure) scored 84 points out of 100 (standard deviation: 13); the perceived usability scored 82 points out of 100 (standard deviation: 12); the learnability scored 91

points out of 100 (standard deviation: 17). The last two factors have been calculated as suggested in (Lewis & Sauro, 2009).

These results shown in Figure 6.10 demonstrate that this gestural interface is suitable for the IVIS application.

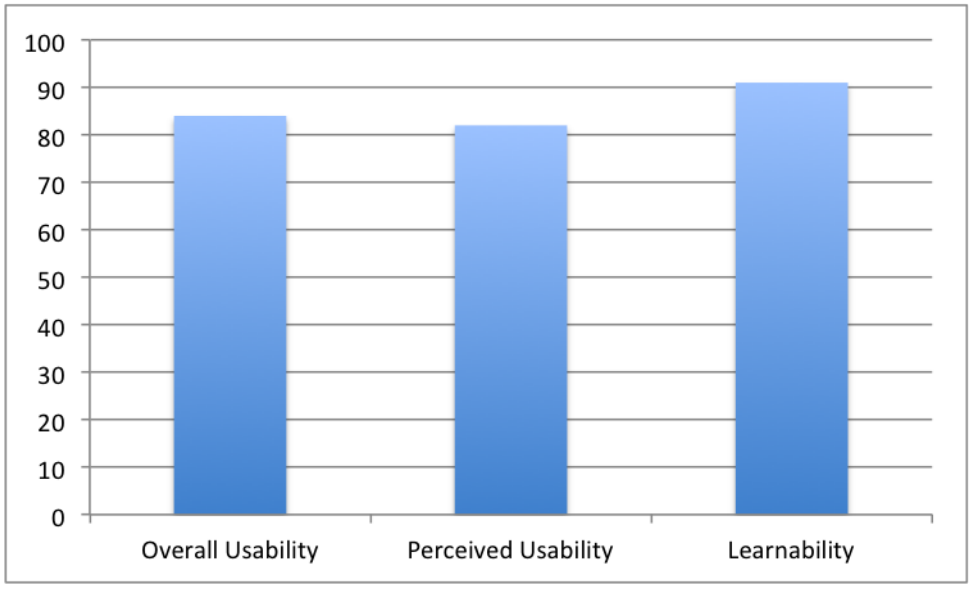


**Figure 6.10 Graph representing the scores obtained with the SUS evaluation.**

### 6.3.3 Implementing the wearable paradigm

Maintaining the same interface design, a second prototype has been developed adding a wearable system and adopting the design principles of the synergistic pervasive paradigm. This system adopts the wearable paradigm that consists in placing the sensors on the user. These sensors capture the electromyographic (EMG) signals generated by the electrical muscles activity. The wearable system sends these data to the IVIS. Finally, the IVIS interprets these data as commands. Sensors were positioned by non-medical personal following Cram's guide (Cram et al., 1998). On the left arm, the sensor used for segmentation was placed on the *Flexor Carpi Ulnaris*. On the right arm we used the electrical activity of the *Extensor Carpi Ulnaris  Flexor Carpi Ulnaris*, *Extensor Digitorum*, *Flexor Carpi Radialis*, and *Palmaris Longus* (Figure 6.11).

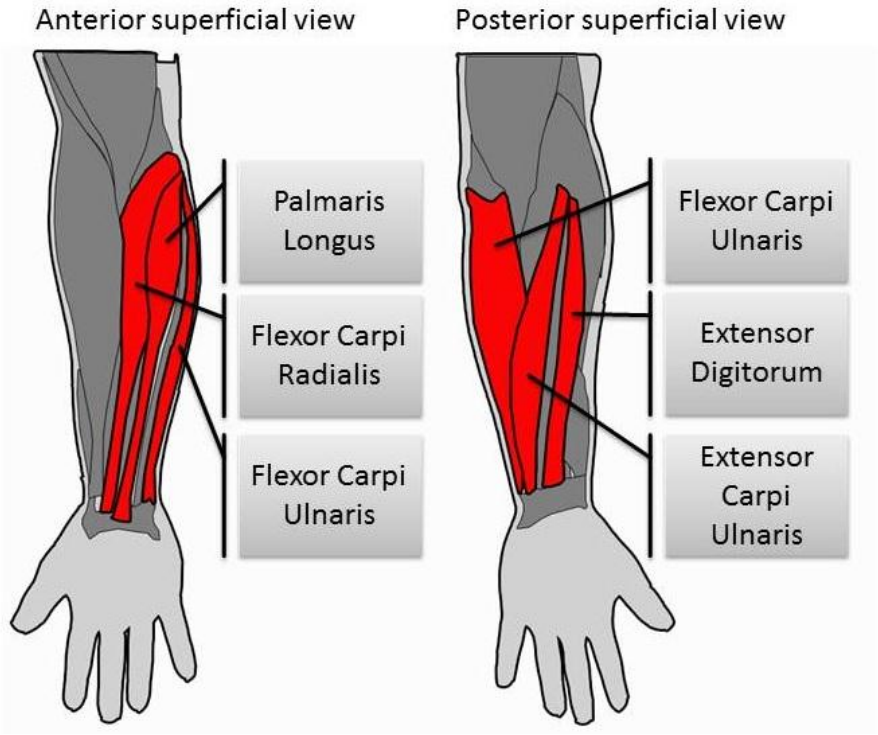


**Figure 6.11 Highlights of the used muscles.**

The possibility to easily integrate the electrodes in clothes or in an armband dictated the choice of the muscles, which excluded the hand muscles. For each sensor, we extract the following features: signals Root Mean Square, Logarithmic Band Power and Mean Absolute Value, for a total of 12 features for the right arm and 3 features for the left. For the setup we used a Marq-Medical MQ16 device.

This second prototype merges the environmental (the pressure sensors on the steering wheel) and the wearable (the EMG sensors on the user's arms) combining the complementary advantages of these two components, as shown in Figure 6.12.

**Figure 6.12 Left: environmental sensors. Right: wearable sensors.**

Users not wearing sensors can interact with the environment according to the "come as you are" design (Wachs et al., 2011). On the other hand, users wearing sensors can exploit them to interact with the system without the need of an augmented steering wheel. Finally, if sensors are present both in the environment and on the user a richer and more accurate interaction can be performed. The environmental and the wearable computing can be designed through eight parameters:

- **Interaction area** is the space in which the user interactions and commands are sensed by the system.
- **Personalization** is the capacity of the system to provide and maintain personalized interactions for the users.
- **Consistency** is the capacity to improve the system thanks to the prolonged, continuous interaction between the human and a computer.
- **Private interaction** and intimate interfaces are "discre[e]te interfaces that allow control of mobile devices through subtlety gestures in order to gain social acceptance" (Costanza et al., 2005).
- **Localized information** is the system feature that specifies how to access the information in a specific location (such as the cockpit, the windshield, or the steering wheel).
- **Localized control** is the system feature that specifies how and where to provide information and commands to the system.

- **Resource availability** is strictly linked to the current technologies adopted for the interaction, e.g., processing power, energy, etc.
- **Resource management** is the system capability to efficiently handle the different available resources. A smart environment, like a vehicle of the next future should deal with heterogeneous sensors, actuators, and multiple users with different tasks and needs.

### 6.3.4 Implementing the synergistic paradigm

As Rhodes et al. claimed, the wearable paradigm is advantageous to facilitate private interaction and the management of personal information; for example, it is simple to define a personalized profile for each user and it is not necessary to provide the user private information to external systems.

The environmental paradigm strong points are the localized information, the localized control and the resource management. Localizing the control and the information in the environment can impose physical constraints. These constraints can be facilitators to the interaction or can improve safety. For example, in this prototype, five pressure sensors are located in specific regions of the external ring to force the user to be compliant with the hands position suggested by the Swiss driving school manual. The system discussed in this paper is depicted in Figure 6.13. The dotted connectors represent loose links.

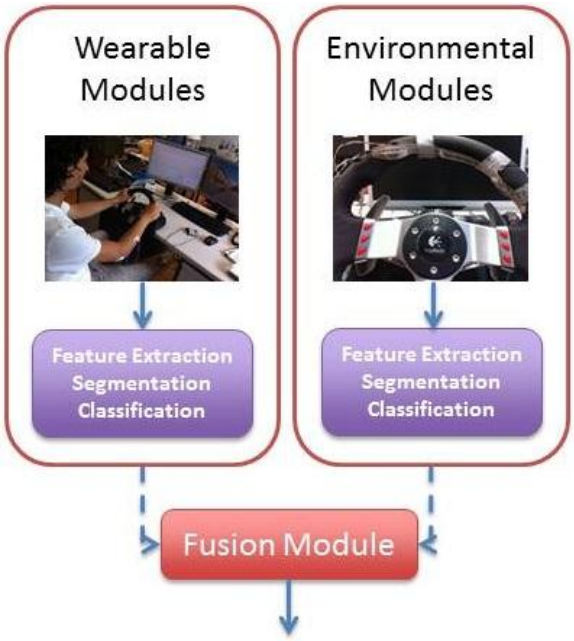


**Figure 6.13 Synergistic paradigm in the car. Sensors are embedded in the steering wheel and worn by the user.**

In order to realize a synergistic paradigm, the fusion of the wearable and pervasive blocks should be accurately designed. The fusion of multiclass classifiers is a critical task in machine learning problems. In particular, the sub-problem of performance measures of multi-class classifiers is still an open research topic (Sokolova & Lapalme, 2009). The fusion of the data coming from the wearable and pervasive classifiers is crucial in order to profitably merge the information and improve the performance of the synergistic system. This schema is based on a late fusion approach and different methodologies to score and weight the results of each class probability are investigated.

The late fusion approach has the advantage to be less affected by changes and modifications in a subsystem and, therefore, facilitate the realization of a synergistic paradigm. The fused decision is selected through the Sum rule. As defined in (Kittler & Alkoot, 2003): “The Sum rule operates directly on the soft outputs of individual experts for each class hypothesis, normally delivered in terms of a posteriori class probabilities. The fused decision is obtained by applying the maximum value selector to the class dependent averages of these outputs.” In another paper, (Kittler et al., 1998) also stated that, generally, “*the*

*combination rule developed under the most restrictive assumptions – the sum rule – outperforms other classifier combination schemes."* In a two-classifier problem the sum rule is reduced to the comparison of the output of the two experts. It is important to note, that the term "expert" encompasses both the classifier and the subsequent weighting process. In this research, a classifier fusion engine has been conceived in order to compare ten different approaches estimating the soft output of the experts. The weighting process is based on confusion matrices and the classifiers soft outputs that have been estimated in a cross-validation phase (performing a k-fold cross-validation on the training set). The following subsections present the weighting processes that have been studied in this experiment; some of them come from the literatures (the Sum Rule and the naive Bayes combination) and are compared with a new technique proposed in this thesis.

#### 6.3.4.1 Sum Rule

This method uses directly the likelihood outputted by the classifiers as weights for the Sum Rule (SR) (Kittler et al., 1998).

#### 6.3.4.2 Naive Bayes combination

The Naive Bayes (NB) combination is very common in the literature as decision rule for the fusion of classifiers because it can be easily extended to more than two classifiers (Kuncheva et al., 2001). It exploits the conditional a posteriori probability $P_j(i|s_j)$ of each classifier that the received gesture belongs to the $i^{th}$ class, given that the $j^{th}$ classifier has assigned that gesture to the $s^{th}$ class ($s_j$). These conditional probabilities can be calculated by the confusion matrix *CM* generated for each classifier during the training phase. Under the assumption that the classifiers are mutually independent, the NB combination calculates the overall a posteriori probability of a class *i* as:

$$W_{Bayesian} = P(i) = \prod_{j=1}^{L} \frac{cm_j(i,s_j)}{\sum_{k=0}^{C} cm_j(k,s_j)}, \quad \text{i=1,..,C} \qquad (1)$$

Where *L* is the number of classifiers, $cm_j(\cdot,\cdot)$ are the confusion matrix elements of the $j^{th}$ classifier, and *C* is the number of classes.

From the NB definition, it is possible to understand that if a classifier emits a soft output, this value is not taken into account by the NB classifier. For this reason, a modified variant of the NB approach has also been included in this experiment; this variant consists in the multiplication of *P(i)* by the soft output of the classifier (e.g., the likelihood in the case of HMMs). This method is denominated with the abbreviation *NBW*.

#### 6.3.4.3 Matthews Correlation Coefficient method

The Matthews Correlation Coefficient (MCC), also known as *φ-coefficient* in the binary case, is an aggregate objective function (AOF) that is used in machine learning as performance measure in the context of multiclass classification (Matthews, 1975). Since this coefficient is stable even if classes are of different sizes, it is adapted to this multi-class problem using a one-vs-all approach (please, refer to (Jurman et al., 2012) for a deeper insight on the use of MCC in machine learning). MCC is defined as:

$$MCC = \frac{(TP * TN - FP * FN)}{\sqrt{(TP + FP) * (TP + TN) * (FP + FN) * (TN + FN)}}$$

Where TP indicates true positive, TN true negative, FP false positive, and FN false negative. This correlation coefficient returns a value between -1 and +1. A value of -1 indicates total disagreement between the classifiers, 0 is for completely random prediction and +1 means total agreement.

The MCC of the aggregate confusion matrix (aggregation of k matrices obtained through the k-fold approach on the training set) is used as weight in the MCC method. The MCC is computed for each gesture class.

To solve potential disagreement between the two classifiers, it is important to introduce in the fusion process the information about the confidence of a classifier in predicting the class guessed by the other classifier. For example, if the first classifier predicts the class $s_1$ and the second the class $s_2$, it is important

to know the performance of the first classifier on the class $s_2$ and of the second classifier on the class $s_1$. In order to deal with this information, the *MCC Conditional* approach (MCC+) has been introduced. Finally, in a configuration with 2 classifiers, the weight of a classification result is computed as the multiplication of the MCC of the predicted class and the MCC of the class predicted by the other classifier:

$$W_i^{j_1} = MCC_{s_1}^{j_1} * MCC_{s_2}^{j_1}$$

Where the notation $MCC_{s_j}^{j}$ indicates the MCC coefficient computed on the confusion matrix of the classifier $j$ predicting the class $s_j$. In particular, $s_1$ is the class predicted by the classifier $j_1$ and $s_2$is the class predicted by the classifier $j_2$. The main drawback of this approach is that it needs to be reformulated in order to work with more than two classifiers.

#### 6.3.4.4 Scheme variant

For all the previous approaches, a variant approach has been developed. This variant takes into account the overall accuracy of the classifiers after a cross-validation step. Multiplying a weight by the classifier overall accuracy brings the fusion algorithm to increase the confidence on the classifier that perform better during the cross-validation phase.

For example, $W_{Bayesian*}$ is computed by multiplying the base approach (i.e., BN) by the overall accuracy. Then:

$$W_{Bayesian*} = W_{Bayesian} * Overall\_Accuracy$$

These modified versions of every method are named with the apex * (e.g., the variant of the MCC using the weighs related to the particular classifier is here called MCC*).

### 6.3.5 Segmentation

For the realization of a synergistic paradigm a critical step is the processing of the segmentation signals. Mainly for two reasons: firstly, from an interaction point of view the segmentation can have an impact on the cognitive load on the user; secondly, having sub-systems working asynchronously, the fusion system can receive gestures that are delayed and it can deal with missing signals. Hence, the maximum possible delay between the outputs of the different classifiers has been fixed at 500ms to be considered as belonging to the same gesture. Otherwise, the gestures are treated separately and the results are not merged. This approach implies a minimum delay (500ms) between the user gestures. To deal with the cognitive load on the user, these two approaches have been compared. The first, called manual segmentation, requires the user to communicate to the system the start and the end of a command gesture. This approach helps the system to understand if a gesture is intended for interaction with the system or it is a manipulation of the steering wheel due to the normal driving task. A simple way to achieve the manual segmentation is to perform a segmentation gesture on the steering wheel with the left hand while the right hand is performing the actual command gesture. The second, called automatic segmentation, implies that the algorithm used to recognize the gesture can automatically detect if the gesture is intended for the IVIS or it is normal steering wheel manipulation for driving.

#### 6.3.5.1 Manual Segmentation

Manual segmentation involves the use of the left hand in order to directly provide the system the “start recognition” and the “stop recognition” commands.

A practical example (used in this prototype) is the following: while the user is squeezing the steering wheel with the left hand, a gesture performed on the steering wheel by the right hand is meant as command and must be recognized.

Figure 6.14 presents the architecture of the system based on manual segmentation.

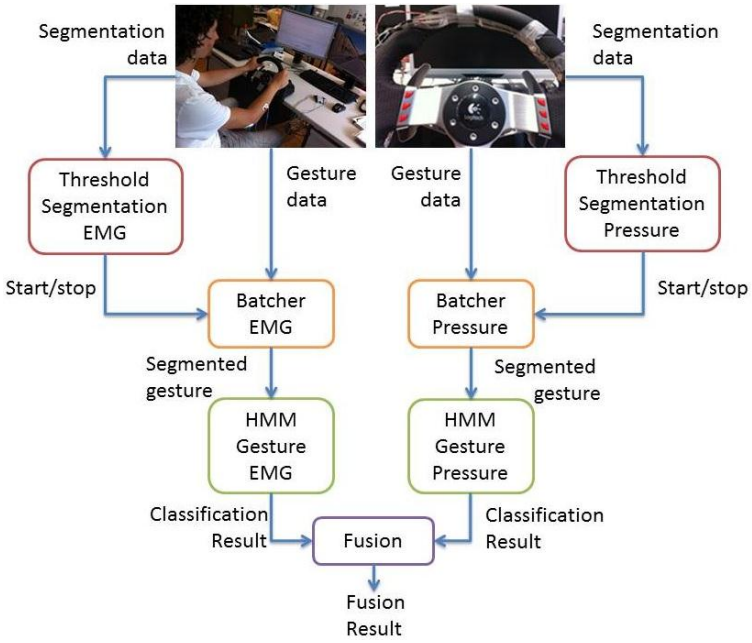


**Figure 6.14 Manual segmentation architecture.**

The sensors provide different types of information flow: segmentation data and gesture data. The information coming from EMG and pressure sensors is analyzed in parallel. The first step involves the processing of the segmentation data. The data are segmented using thresholds that are fixed in a user-dependent calibration phase. In particular, a hysteresis-based approach has been adopted, in which the activation and deactivation thresholds are calculated taking into account the standard deviation of the user's signals when she/he is driving or she/he is performing the Squeeze (see Figure 6.15). Therefore, the segmentation is enabled when the signal (the root mean square for the EMG and the raw data for the pressure sensor) exceeds the activation threshold, and is disabled when the signal drops below the deactivation threshold. The Batcher modules start to accumulate the data on the start signal. With the stop signal, the collected information is sent to the classification algorithm.

The HMM classifiers were configured with 4 hidden states with forward topology and implementing the Baum-Welch algorithm to find the unknown parameters.

Finally, the outputs of the HMMs are merged in the fusion module.

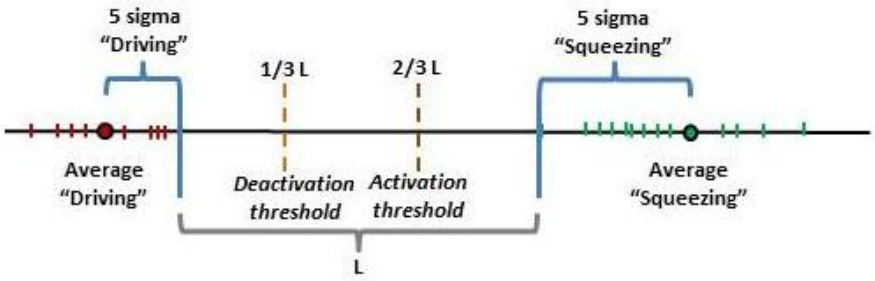


**Figure 6.15 Manual segmentation - Threshold computation.**

#### 6.3.5.2 Automatic Segmentation

The automatic segmentation uses the same signals for gesture recognition and segmentation. Practically, it involves a reduced cognitive load on the user who has to perform only right hand gestures on the steering wheel. The architecture presented in Figure 6.16 is slightly different from the one presented in Figure 6.14. In this configuration only gesture data are used. In order to spot the start and the end of a gesture, the system analyzes the data flows using a windowing approach.

For each window, a Support Vector Machine (SVM) classifier estimates if the user is performing a gesture or is simply driving. The SVM algorithm uses a Gaussian kernel in which an appropriate value of sigma is obtained using the approach explained in (Caputo et al., 2002). Since the segmentation events are triggered after a window of data, there is the risk of losing important information. The buffered batcher modules allow queuing the data and, therefore, avoiding this loss. The following part of the schema is the same as in the manual segmentation approach, with HMMs classifiers analyzing the data for gesture recognition.

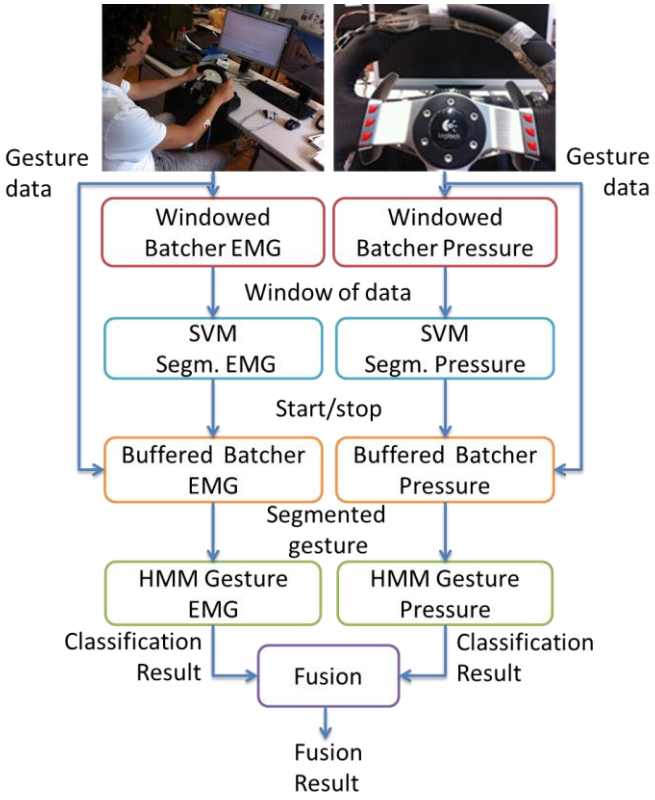


**Figure 6.16 Automatic segmentation architecture.**

Three different segmentation strategies have been implemented in order to analyze the influence of the segmentation on the synergy. The OR strategy considers as valid results the contribution of both single and coupled classifiers (as the logic operator OR). The AND strategy takes into account only gestures simultaneously detected by the two classifiers. The ADAPTIVE strategy opportunistically switches between the OR and the AND strategies according to the following rule using the confusion matrices of the segmentation:

If the classifier $j_1$ segments a gesture s ignored by the classifier $j_2$, and if $FP_{J_1}(s) > FN_{J_2}(s)$ then the detected gesture is considered a *FP*, i.e., incorrectly segmented. Otherwise, the gesture is considered as correctly segmented.

## 6.4 Test

The prototype has been tested with 9 users (1 female), aged 24-31. First, there was a familiarization session of about five minutes, and then there was the calibration phase. For this phase, a user-dependent solution has been adopted, which has the advantage to generally perform better than user-independent solutions. The drawback is that the system will always require a calibration and a training phase on the first usage. In order to mitigate these factors, the prototype has been designed to need reduced training data. The calibration phase aim is to detect the average strength applied by the user on the steering wheel and the average electrical activity of the muscles. Such signals are user-dependent, depending on the user driving habits and muscular development.

The gesture taxonomy for this second prototype has been extended to five micro-gestures in order to include a new command: pause. The five gestures are: *Squeeze*, *Up*, *Down*, *Tap*, and *Push* (as shown in Figure 6.17). The *Squeeze*, *Up*, *Down*, and *Tap* remain the same gestures as designed for the first prototype, the *Push* consists of a radial deviation while holding the steering wheel.

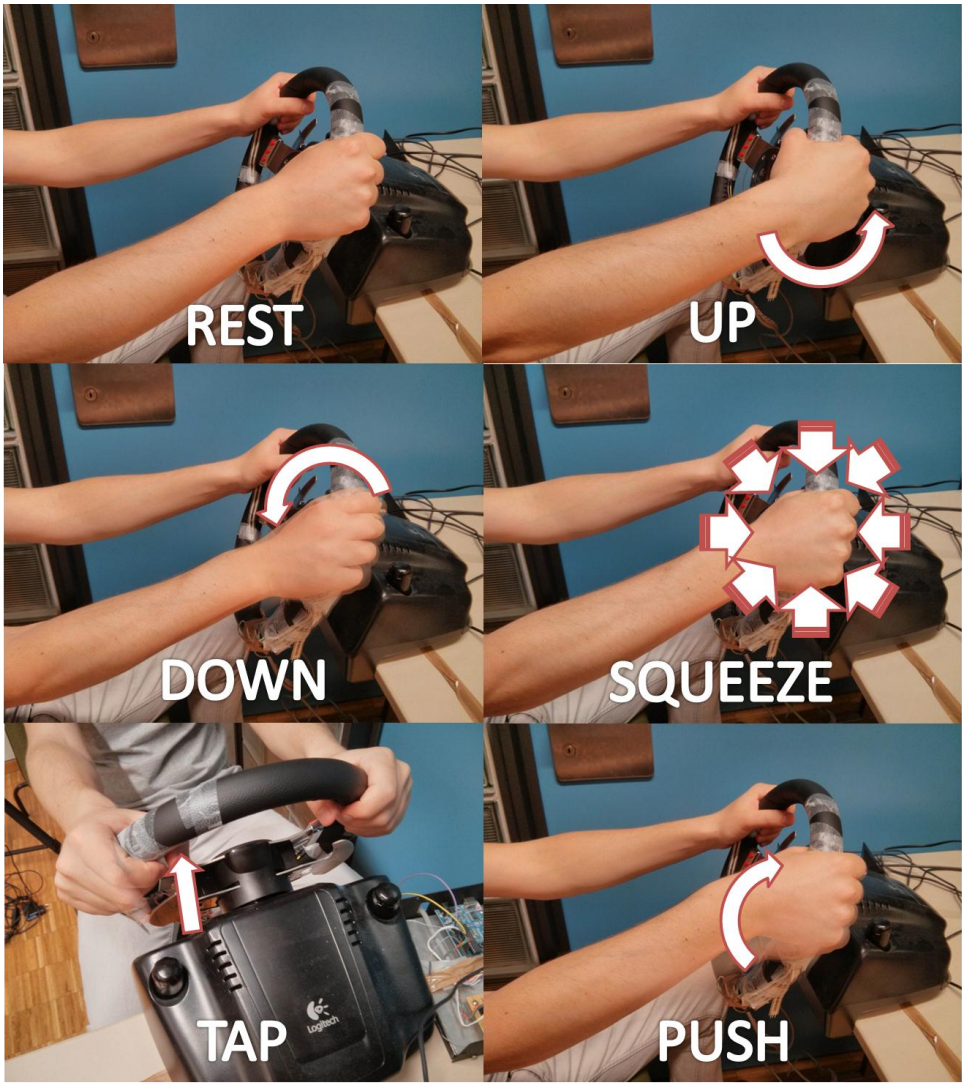


**Figure 6.17 The micro-gesture vocabulary for the evaluation with the synergistic paradigm.**

The mapping between functions and gestures was:

- Squeeze -> Stop
- Up -> Next
- Down -> Previous
- Tap -> Play
- Push -> Pause

Each user was asked to perform 30 gestures for each of the five gestures (*Squeeze*, *Up*, *Down*, *Tap*, and *Push*) in a random order. During the execution of

the gestures, the user was asked also to perform normal driving actions on the steering wheel: turn left, right, accelerate, brake or stay in a rest position.

The acquisition protocol consists of the simultaneous acquisition of data for the manual and automatic segmentation. The users had to squeeze the external ring of the steering wheel with the left hand to perform the manual segmentation. This information was then elaborated only by the manual segmentation modules. The whole acquisition process was performed in two sessions of about 15 minutes in order to allow the users to rest.

The 70% of the gestures were used as training and cross-validation sets; the 30% of the gestures were used as test set. It has been applied a k-fold (k=10) cross-validation on the training in order to calculate the confusion matrix and to select the weights for the fusion module.

The goal of this second prototype is to provide data to evaluate three aspects of the proposed synergistic paradigm: the interaction opportunities and limitations, the sensor fusion methodologies and the gesture segmentation.

Since the interaction opportunities are strictly related to the results of the fusion and segmentation steps, a quantitative evaluation of data fusion and gesture segmentation strategies is presented. Subsequently, a qualitative discussion about the interaction features of the synergistic paradigm is provided at the end of this section.

This analysis is based on the comparison the F1-score and the accuracy in the classification. The F1 score is defined as:

$$F_1 = 2 * \frac{precision * recall}{precision + recall}$$

Where:

$$precision = \frac{TP}{TP + FP}$$

$$recall = \frac{TP}{TP + FN}$$

The accuracy is defined as the number of TP divided the total number of gestures.

Table 6.1 presents these values for the different classifier fusion methods and segmentation strategies.

**Table 6.1 Fusion results (μ and σ) F1 score and accuracy. In red the best single classifier; in bold the best fusion methods.**

| (μ\|σ) | Manual Segment. OR | | Automatic Segment. OR | | Manual Segment. AND | | Automatic Segment. AND | | Manual Segment. ADAPTIVE | | Automatic Segment. ADAPTIVE | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Method** | **F1** | **A** | **F1** | **A** | **F1** | **A** | **F1** | **A** | **F1** | **A** | **F1** | **A** |
| **Pressure** | 0.72\|0.16 | 0.79\|0.14 | 0.66\|0.20 | 0.76\|0.17 | 0.72\|0.17 | 0.76\|0.15 | 0.65\|0.15 | 0.76\|0.13 | 0.72\|0.16 | 0.76\|0.13 | 0.65\|0.16 | 0.76\|0.12 |
| **EMG** | 0.85\|0.12 | 0.86\|0.13 | 0.73\|0.19 | 0.75\|0.19 | 0.85\|0.12 | 0.85\|0.13 | 0.75\|0.19 | 0.74\|0.19 | 0.85\|0.11 | 0.85\|0.12 | 0.74\|0.19 | 0.72\|0.19 |
| **SR** | 0.79\|0.13 | 0.85\|0.11 | 0.70\|0.15 | 0.84\|0.09 | 0.77\|0.18 | 0.72\|0.24 | **0.78\|0.12** | **0.73\|0.14** | 0.81\|0.12 | 0.82\|0.12 | **0.74\|0.11** | **0.77\|0.11** |
| **NB** | 0.79\|0.14 | 0.84\|0.10 | 0.71\|0.15 | 0.84\|0.08 | 0.74\|0.21 | 0.70\|0.26 | 0.76\|0.12 | 0.70\|0.14 | 0.78\|0.11 | 0.79\|0.11 | 0.71\|0.16 | 0.74\|0.15 |
| **NBW** | 0.80\|0.14 | 0.85\|0.10 | 0.71\|0.15 | 0.84\|0.08 | 0.75\|0.21 | 0.71\|0.27 | 0.77\|0.13 | 0.71\|0.14 | 0.80\|0.12 | 0.80\|0.12 | 0.71\|0.15 | 0.74\|0.15 |
| **MCC** | 0.81\|0.14 | 0.87\|0.09 | **0.72\|0.14** | **0.86\|0.09** | 0.80\|0.20 | 0.75\|0.26 | **0.78\|0.13** | **0.73\|0.15** | 0.84\|0.11 | 0.85\|0.10 | **0.74\|0.13** | **0.77\|0.12** |
| **MCC+** | 0.81\|0.15 | 0.87\|0.10 | 0.69\|0.16 | 0.83\|0.14 | 0.78\|0.20 | 0.73\|0.25 | **0.78\|0.16** | 0.72\|0.17 | 0.83\|0.12 | 0.85\|0.11 | 0.73\|0.16 | 0.76\|0.15 |
| **SR*** | 0.79\|0.13 | 0.85\|0.11 | 0.70\|0.15 | 0.84\|0.09 | 0.77\|0.18 | 0.72\|0.24 | **0.78\|0.12** | **0.73\|0.14** | 0.81\|0.12 | 0.82\|0.12 | **0.74\|0.11** | **0.77\|0.11** |
| **NB*** | 0.79\|0.14 | 0.84\|0.10 | 0.71\|0.15 | 0.84\|0.08 | 0.74\|0.21 | 0.70\|0.26 | 0.76\|0.12 | 0.70\|0.14 | 0.78\|0.11 | 0.79\|0.11 | 0.71\|0.16 | 0.74\|0.15 |
| **NBW*** | 0.80\|0.14 | 0.85\|0.10 | 0.71\|0.15 | 0.85\|0.08 | 0.75\|0.21 | 0.71\|0.27 | 0.77\|0.13 | 0.71\|0.14 | 0.80\|0.12 | 0.80\|0.12 | 0.71\|0.15 | 0.74\|0.15 |
| **MCC*** | **0.83\|0.15** | **0.89\|0.10** | **0.72\|0.14** | **0.86\|0.09** | 0.80\|0.21 | 0.75\|0.26 | **0.78\|0.15** | **0.73\|0.16** | **0.85\|0.13** | **0.86\|0.12** | **0.74\|0.14** | **0.77\|0.13** |
| **MCC+*** | 0.82\|0.15 | 0.88\|0.10 | 0.71\|0.18 | 0.84\|0.15 | **0.81\|0.21** | **0.76\|0.27** | 0.77\|0.17 | 0.72\|0.18 | **0.85\|0.13** | **0.86\|0.12** | 0.72\|0.16 | 0.76\|0.15 |

Pressure and EMG rows present the results of the single classifiers. It is possible to observe that in this configuration the EMG performs generally better than the pressure sensors. However, the synergistic paradigm can perform equal or better than the best stand-alone classifier.

In particular, the best results have been achieved with the ADAPTIVE segmentation strategy and the MCC* fusion method. Such configuration performs better of the other fusion methods and always equal or better than the best standalone classifier.

The proposed fusion approaches are independent from the classifiers typology and can be extended to classifiers with crisp outputs (the Sum rule is then reduced to a Vote rule). However, when the fusion method uses probabilistic weights, it is necessary to use the same type of classifier for the environmental and wearable components in order to guarantee the mathematical significance of the fusion.

Finally, the manual segmentation performs about 6% better than the automatic segmentation. In order to understand which segmentation approach should be chosen in an interaction system, the trade-off between the cognitive load on the user and the effect of a lower accuracy of the gesture recognition should be evaluated.

The quantitative evaluation on this dataset shows good performances in terms of accuracy and F1-score. However, this second prototype allowed investigating also more qualitative features of a synergistic paradigm in the context of the interaction in a car. Referring to the aforementioned eight design parameters, in this section it is presented how they influence the implementation of this opportunistic system that adopts the synergistic paradigm.

A direct consequence of the wearable paradigm is that the interaction area is no more limited to some spots in the car but can be extended to the whole car. Once the communication between the wearable and the environmental systems is established, it is possible to control the IVIS everywhere in the car.

Gestures can be user dependent. Personalized configurations and profiles can be shared from the wearable system to the environmental one. In addition, a wearable system can be designed to share the user information with an unknown system. For example, in a car-sharing scenario, the wearable

information of a user can be used to configure a new car automatically: binding the personalized gesture vocabulary to the system controls.

With a synergistic system an easy way to increase consistency is to online adapt the weights used in the fusion module to achieve better performances. For example, it is possible to penalize or award a classifier directly decreasing or increasing the weights used by the SR or the MCC methods.

The car is an interaction milieu that is generally considered as private. Therefore, in this specific context the needs of private interaction and intimate interfaces are reduced, even though the wearable components represent a good solution for this problematic.

The environmental component of a synergistic system can provide localized information to the driver taking into account the specificities of the car. For example, the windshield can be used to display information to the user that does not need to move the gaze from the road conserving the safety of the driving. As mentioned before, the five pressure sensors were positioned in specific regions of the external ring of the steering wheel to have localized controls helping the driver to keep the hands on the steering wheel while interacting with the IVIS.

The environmental sensing can be directly integrated in the car exploiting the existing processing and power resources. A synergistic paradigm allows the wearable system to take advantage of the vehicle resources availability. In fact, even if the wearable components still require energy to sense the information, the processing can be deployed at environmental side. The downside of this approach is that the transmission of the information can be highly demanding in term of energy and may slow down the whole processing system. Therefore, accurate analyses should be performed case-by-case.

The presence of different gesture lexicons for the wearable and the environmental paradigm can be treated as resource management. Commands linked to secondary tasks can be integrated as wearable components and made available for the driver as well as the passengers of the vehicle; on the other hand, primary commands, that can affect passengers' safety, can be made

accessible only to the driver by localizing the control in a particular spot of the vehicle.

## 6.5 Summary

Chapter 3 introduced the framework for the description of the context awareness with reference to the relation between the user and the objects. Chapter 4 described the application of the framework and presented a novel technique for the view-invariant gesture recognition. Chapter 5 further examined the user activity and proposed a novel approach for the human activity recognition. Chapter 6 introduced the system status as an important factor for the description of the context. The system status has been treated with reference to the different paradigms that can be adopted implementing a gestural interface of the ubiquitous computing era. In particular, this chapter introduced the concept of synergistic paradigm, which is the combination of the wearable and environmental paradigms in an opportunistic way. Mixing these two paradigms allows improving the user experience because it merges the advantages coming from these two paradigms. These advantages are related to the features of privacy, personalization, localized information, localized control, and resource management, as previously explained. Moreover, the presence of both systems can improve the gesture recognition through the implementation of a late decision fusion algorithm. In this thesis, different fusion methods were tested; nonetheless, a comparison among the different methods is conducted in order to quantify the improvement of the gesture recognition accuracy. The best overall accuracy has been achieved using a variant of the MCC method that has been introduced in this thesis. Moreover, the synergistic paradigm has been designed to grant the functioning of the whole system even if one of the two components (i.e., wearable and environmental) is missing. This feature provides the maximum freedom to the user, who can move in different environments maintaining a seamless interaction experience.

# Chapter 7: Conclusions and Future Work

## 7.1 Conclusions

The human-computer relationship profoundly changed in the last two decades. In fact, the current wave of the human-computer interaction has arrived and is called ubiquitous computing era, which means that every user owns many digital devices that are always connected. The pervasiveness of technology allows easier access to information and the possibility to stay in touch with people but at the same time it overwhelms the user compromising the interaction experience. In this scenario, the human-computer interaction design plays a crucial role in order to provide usable technology. In fact, currently the user is put at the center of the system design in order to provide interfaces that can be intuitive, easy to learn and enjoyable to use. A main trend in this direction is the adoption of the natural means of human communication, namely, voice and gestures. In particular, gestures represent the best modality for the interaction with smart environments and for this reason many studies have focused on the development of gestural interfaces. Unfortunately, this rapid popularity gained by the gestural interaction has led to a wild, incoherent and fragmented development of this field. This is principally due to a lack of common guidelines for gestural interface design and development. In particular, there has been little work that deals with the problem of context-aware gestural interaction with smart environments. For this reason, this thesis aimed at developing a novel framework for context-aware gestural interaction in the smart environments of the ubiquitous computing era, which has been achieved. In fact, this thesis has

introduced a novel high-level framework for the design and development of context-aware gestural interfaces for the interaction with ubiquitous computing environments. The framework is based on the DIKW pyramid for the information organization and on a specific spatial model for interaction description. It is suitable for the development of interfaces that can reduce the ambiguity of gestures with reference to the interaction context. Hence, a novel approach for gestural interface design based on this framework is also presented. The approach integrates the concept of functional gestures and reduces the number of gestures to augment the interface usability. Moreover, the framework has taken into account the contextual information associated with the interactive entities present in the environment. The analysis and application of the framework have been conducted to simplify the model comprehending the user status, the system status and the description of their interaction. In particular, the user status information is based on the human activity recognition; the system status is based on the novel synergistic paradigm and the dynamic management of the wearable and environmental subsystems.

## 7.2 Contributions

The main contributions of this research consist of the development of a novel high-level framework for context-aware gestural interaction with smart environments. This framework is strictly linked to the introduction of a novel approach for the gesture interface design based on the concept of functional gestures. The framework in conjunction with the novel approach for the functional gesture design allows developing more natural interfaces, which means easy to learn and to use. In fact, this approach enables designers to reduce the number of gestures introducing the concept of generic functions rather than specific commands. For instance, “turn on the TV” or “turn on the lamp” are two different commands and a classic one-to-one mapping between gestures and commands would involve implementing an interface with two different gestures. The proposed approach allows abstracting the two commands

as a single function, in this example “turn on”, that can have different target objects. In this case, it is possible to implement an interface that has the same gesture for both commands: “turn on the TV” and “turn on the lamp”. This is only possible thanks to the opportune use of the context information to select the target, therefore, to disambiguate the meaning of a specific gesture. The framework also allows for a general organization of the context information that facilitates the reuse of elaborated information on different levels of complexity.

The development of the proof-of-concept prototypes was not limited to only represent an example of application of the framework. Indeed, the development of the proof-of-concept gave the opportunity for the development of a novel technique for the view-invariant recognition of 3D gestures. In particular, this thesis presents two algorithms for the deictic and dynamic gesture recognition utilizing very recent technology, i.e., the Microsoft Kinect cameras. The deictic gesture recognition algorithm is based on a 3D model and a decision tree for the tracking of the arm rotations with 3D temporal trajectories and HMM classifiers. The experiments conducted with real users have assessed the high recognition accuracy of these algorithms independently of the user rotation on 180°. Moreover, other application scenarios have been implemented and tested with users to investigate the user experience with focus on the usability of the proposed interfaces. The usability tests, based on the System Usability Scale questionnaires, have achieved encouraging results for the implementation of this kind of gestural interfaces.

The context includes the information concerning the user status. In the frame of this research, the user status is based on the human activity recognition. A novel technique based on an innovative application of the EMG for the human activity recognition has been presented. The technique is based on a simple preprocessing of the EMG signals and, after the windowing, the recognition is demanded to the HMM classifier. It has been tested with real users performing five specific activities (walking, running, cycling, standing and sitting). The results show very good recognition accuracy; moreover, additional tests have been

conducted to provide a table that can help developers to optimize the system in finding the best compromise between the number of muscles sensed and the desired activity recognition accuracy. Another very important contribution generated by this investigation is the identification of the most important muscles concerned by the selected activities. In fact, through a comprehensive literature review and some preliminary empirical tests, it has been possible to provide a list of four muscles that play a crucial role for the recognition of the five activities.

Last but not least, a novel paradigm (called synergistic paradigm) for the ubiquitous computing has been presented in this research thesis. It allows combining the advantages of the wearable and environmental paradigms. In fact, the synergistic paradigm involves developing a system that is composed of a wearable subsystem and an environmental subsystem; these two subsystems are dynamically managed. This means that the user is able to interact with the smart environment as long as at least one subsystem is available. If both subsystems are available, the synergistic paradigm includes the implementation of a fusion engine that allows improving the gesture recognition accuracy. In fact, it ensures that the gesture recognition accuracy is always no lower than the best accuracy obtained with a single subsystem. Moreover, it provides all the advantages related to the features of privacy, personalization, localized information, localized control, and resource management. The sum of all these improvements provided by the implementation of the synergistic paradigm allows enhancing the user experience during the interaction with the gestural interface.

## 7.3 Research limitations

The concept of context is very broad and integrates many different aspects of human-computer interaction. This research has introduced a novel high-level framework to integrate all the information associated to the human-environment interaction of the ubiquitous computing era. However, the investigation conducted on the application of the framework has been limited to

specific aspects, which are: the interaction description, the user status and the system status. The interaction description based on the gesture recognition is limited to the deictic and dynamic gestures. The user status is limited to the human activity recognition concerning five specific physical activities. The system status is restricted to the implementation of the novel synergistic paradigm of the pervasive computing.

This framework allows for the classification of every type of context information. Unfortunately, in this research it has been adopted in specific scenarios and only an extensive use of the framework can lead to a standard for the opportune organization of the elaborated information of the different levels of the DIKW pyramid.

## 7.4 Future work

In this emerging domain, there are a lot of investigations to perform yet. Concerning the research conducted in this thesis, further work can be done for each contribution. The framework has been conceived to be general and to include all the information concerning the context. However, the application of the framework is limited to the resolution of ambiguity to reduce the number of gestures present in the interface vocabulary. For this reason it is not possible to investigate the classification of all the information in the framework. This should be performed with the development of new features and new interfaces. This work would be facilitated if other researchers could adopt the framework for the development of their interfaces sharing their levels of information elaboration. In fact, the different levels of the information can be presented on the DIKW pyramid and treated as services in the actual system architecture. The standardization of the information elaboration could be an important added value to the framework.

Another aspect that could be improved is the technique for view invariant gesture recognition; indeed, it could be extended to include other types of gestures, different from the already implemented deictic and dynamic gestures.

For example, the new algorithms could enable the view-invariant recognition of static gestures as the hand postures. This is a particularly challenging research topic because the human hand is highly articulated and deformable, and that makes it very hard to recognize the same posture from different points of view (Wu & Huang, 2000). However, including also the view-invariant static gesture recognition could augment the interaction possibilities and the richness of the gesture taxonomies. The latter could also augment the affordance of the gestural interface because it could give to the designers more freedom during the gesture design process.

With reference to the user status, the physical activity recognition presented in this thesis has showed very promising results for this technology that was not used before for this purpose. However, further tests can be conducted to understand whether the muscle fatigue can compromise the gesture recognition. Moreover, the natural evolution of this technology could be the integration of the sensors in garments. In fact, intelligent textiles that allow the EMG signal detection do exist and the integration of electronics in garments has a promising future. This could lead to the creation of clothes that could be aware of the user's physiological condition in order to not only provide opportune services but also to monitor the user's health. This development involves further research about the signal degradation with the use of electronic textiles instead of wet electrodes.

Another important area of future work can be the implementation of the highest level of the framework that has not been addressed in this thesis. This level concerns the negotiation between the user and the smart environment. In fact, the interaction between the smart environment and the user should be similar to the one that happens between humans. This means that the smart environment should not always ask for confirmation nor act proactively. The interaction should be like a dialog during which the system should acquire information about the user in order to understand his/her intention quickly. In fact, to build and to maintain a long-term human-computer interaction requires

some kind of natural conversational interface. This does not mean that the interaction should be vocal but that the system should be reactive to the user intervention in every moment and try to accommodate possible requests. So, the proactive action can be started but it should also be easily interrupted if the user wishes to. The system can also propose some actions while considering the current context and asking for permission if necessary. If the user ignores it, the system should not be too invasive. This kind of interaction is very complex since it is very similar to human relationships and it requires further research in the field of artificial intelligence.

# References

Abowd, G. D. (2012, September). What next, ubicomp?: celebrating an intellectual disappearing act. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing* (pp. 31-40). ACM.

Akers, D. L. (2007, April). Observation-based design methods for gestural user interfaces. In *CHI'07 Extended Abstracts on Human Factors in Computing Systems* (pp. 1625-1628). ACM.

Angelini, L., Caon, M., Carrino, S., Bergeron, L., Nyffeler, N., Jean-Mairet, M., & Mugellini, E. (2013, September). Designing a desirable smart bracelet for older adults. In *Proceedings of the 2013 ACM conference on Pervasive and ubiquitous computing adjunct publication* (pp. 425-434). ACM.

Annett, M., Grossman, T., Wigdor, D., & Fitzmaurice, G. (2011, October). Medusa: a proximity-aware multi-touch tabletop. In *Proceedings of the 24th annual ACM symposium on User interface software and technology* (pp. 337-346). ACM.

Arun, K. S., Huang, T. S., & Blostein, S. D. (1987). Least-squares fitting of two 3-D point sets. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (5), 698-700.

Ashford, S., & De Souza, L. (2000). A comparison of the timing of muscle activity during sitting down compared to standing up. *Physiotherapy Research International*, *5*(2), 111-128.

Augusto, J. C. (2010). Past, present and future of ambient intelligence and smart environments. In *Agents and Artificial Intelligence* (pp. 3-15). Springer Berlin Heidelberg.

Bach, K. M., Jæger, M. G., Skov, M. B., & Thomassen, N. G. (2009, September). Interacting with in-vehicle systems: understanding, measuring, and evaluating attention. In *Proceedings of the 23rd British HCI Group Annual Conference on People and Computers: Celebrating People and Technology* (pp. 453-462). British Computer Society.

Badler, N. I., & Smoliar, S. W. (1979). Digital representations of human movement. *ACM Computing Surveys (CSUR)*, *11*(1), 19-38.

Bakker, S., Antle, A. N., & Van Den Hoven, E. (2012). Embodied metaphors in tangible interaction design. *Personal and Ubiquitous Computing*, *16*(4), 433-449.

Baudel, T., & Beaudouin-Lafon, M. (1993). Charade: remote control of objects using free-hand gestures. *Communications of the ACM*, *36*(7), 28-35.

Bellinger, G., Castro, D., & Mills, A. (2004). Data, Information, Knowledge, and Wisdom. Available at: www.systems-thinking.org/dikw/dikw.htm Site accessed on 26 June 2014

Ben Soltana, W., Ardabilian, M., Chen, L., & Ben Amar, C. (2010, August). Adaptive Feature and Score Level Fusion Strategy Using Genetic Algorithms. In*Pattern Recognition (ICPR), 2010 20th International Conference on* (pp. 4316-4319). IEEE.

Bettini, C., Brdiczka, O., Henricksen, K., Indulska, J., Nicklas, D., Ranganathan, A., & Riboni, D. (2010). A survey of context modelling and reasoning techniques. *Pervasive and Mobile Computing*, *6*(2), 161-180.

Bickmore, T. W., & Picard, R. W. (2005). Establishing and maintaining long-term human-computer relationships. *ACM Transactions on Computer-Human Interaction (TOCHI)*, *12*(2), 293-327.

Bolt, R. A. (1980). *"Put-that-there": Voice and gesture at the graphics interface*(Vol. 14, No. 3, pp. 262-270). ACM.

Bolton, E. B. (2001). *About IFAS leadership development: Leaders can communicate*. University of Florida Cooperative Extension Service, Institute of Food and Agriculture Sciences, EDIS.

Bragdon, A., Nelson, E., Li, Y., & Hinckley, K. (2011, May). Experimental analysis of touch-screen gesture designs in mobile environments. In*Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 403-412). ACM.

Brooke, J. (1996). SUS-A quick and dirty usability scale. *Usability evaluation in industry*, *189*, 194.

Bub, D. N., Masson, M. E., & Cree, G. S. (2008). Evocation of functional and volumetric gestural knowledge by objects and words. *Cognition*, *106*(1), 27-58.

Budde, M., Berning, M., Baumgärtner, C., Kinn, F., Kopf, T., Ochs, S., ... & Beigl, M. (2013, September). Point & control--interaction in smart environments: you only click twice. In *Proceedings of the 2013 ACM conference on Pervasive and ubiquitous computing adjunct publication* (pp. 303-306). ACM.

Burke, E. (2002). *Serious cycling*. Human kinetics.

Cabral, M. C., Morimoto, C. H., & Zuffo, M. K. (2005, October). On the usability of gesture interfaces in virtual reality environments. In *Proceedings of the 2005 Latin American conference on Human-computer interaction* (pp. 100-108). ACM.

Caputo, B., Sim, K., Furesjo, F., & Smola, A. (2002, December). Appearance-based Object Recognition using SVMs: Which Kernel Should I Use?. In *Proc of NIPS workshop on Statistical methods for computational experiments in visual processing and computer vision, Whistler* (Vol. 2002).

Carrino, S., Mugellini, E., Khaled, O. A., & Ingold, R. (2011). ARAMIS: toward a hybrid approach for human-environment interaction. In *Human-Computer Interaction. Towards Mobile and Intelligent Interaction Environments* (pp. 165-174). Springer Berlin Heidelberg.

Carrino, S., Péclat, A., Mugellini, E., Abou Khaled, O., & Ingold, R. (2011, November). Humans and smart environments: a novel multimodal interaction approach. In *Proceedings of the 13th international conference on multimodal interfaces* (pp. 105-112). ACM.

Castle, R. O., & Murray, D. W. (2009, October). Object recognition and localization while tracking and mapping. In *Mixed and Augmented Reality, 2009. ISMAR 2009. 8th IEEE International Symposium on* (pp. 179-180). IEEE.

Castle, R., Klein, G., & Murray, D. W. (2008, September). Video-rate localization in multiple maps for wearable augmented reality. In *Wearable Computers, 2008. ISWC 2008. 12th IEEE International Symposium on* (pp. 15-22). IEEE.

Chai, X., Li, G., Chen, X., Zhou, M., Wu, G., & Li, H. (2013, October). VisualComm: a tool to support communication between deaf and hearing persons with the Kinect. In *Proceedings of the 15th International ACM SIGACCESS Conference on Computers and Accessibility* (p. 76). ACM.

Chen, H., Perich, F., Finin, T., & Joshi, A. (2004, August). Soupa: Standard ontology for ubiquitous and pervasive applications. In *Mobile and Ubiquitous Systems: Networking and Services, 2004. MOBIQUITOUS 2004. The First Annual International Conference on* (pp. 258-267). IEEE.

Chen, Y., Chen, R., Chen, X., Chen, W., & Wang, Q. (2011, September). Wearable electromyography sensor based outdoor-indoor seamless pedestrian navigation using motion recognition method. In *Indoor Positioning and Indoor Navigation (IPIN), 2011 International Conference on* (pp. 1-9). IEEE.

Cipolla, R., & Hollinghurst, N. J. (1996). Human-robot interface by pointing with uncalibrated stereo vision. *Image and Vision Computing*, *14*(3), 171-178.

Cook, D., & Das, S. (2004). *Smart environments: technology, protocols and applications* (Vol. 43). John Wiley & Sons.

Cook, J., Cox, M., Chandran, V., & Sridharan, S. (2007). Robust 3D face recognition from expression categorisation. In *Advances in Biometrics* (pp. 271-280). Springer Berlin Heidelberg.

Cooper, H., Holt, B., & Bowden, R. (2011). Sign language recognition. In *Visual Analysis of Humans* (pp. 539-562). Springer London.

Corballis, M. C. (2002). *From hand to mouth: The origins of language*. Princeton University Press.

Cosnier, J. (1982). 4. Communications et langages gestuels. In J. Cosnier, J. Coulon, J. Berrendonner, & C. Orecchioni (Eds*.), Les Voies du langage: Communications verbales, gestuelles, et animales*. (pp. 255-303). Paris: Dunod

Costanza, E., Inverso, S. A., & Allen, R. (2005, April). Toward subtle intimate interfaces for mobile devices using an EMG controller. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 481-489). ACM.

Coutaz, J., Crowley, J. L., Dobson, S., & Garlan, D. (2005). Context is key.*Communications of the ACM*, *48*(3), 49-53.

Cram, J. R., Kasman, G. S., & Holtz, J. Introduction to surface electromyography. 1998. *Gaithersburg, Marland: Aspen publishers Inc*.

Crowley, J. L., Coutaz, J., Rey, G., & Reignier, P. (2002). Perceptual components for context aware computing. In *UbiComp 2002: Ubiquitous Computing* (pp. 117-134). Springer Berlin Heidelberg.

Dang, N. T. (2007, March). A survey and classification of 3D pointing techniques. In *Research, Innovation and Vision for the Future, 2007 IEEE International Conference on* (pp. 71-80). IEEE.

de Carvalho Correia, A. C., de Miranda, L. C., & Hornung, H. (2013). Gesture-based interaction in domotic environments: State of the art and HCI framework inspired by the diversity. In *Human-Computer Interaction–INTERACT 2013* (pp. 300-317). Springer Berlin Heidelberg.

Delaye, A., Sekkal, R., & Anquetil, E. (2011, February). Continuous marking menus for learning cursive pen-based gestures. In *Proceedings of the 16th international conference on Intelligent user interfaces* (pp. 319-322). ACM.

Dey, A. K., Abowd, G. D., & Salber, D. (2001). A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications.*Human-computer interaction*, *16*(2), 97-166.

Do, J. H., Jung, S. H., Jang, H., Yang, S. E., Jung, J. W., & Bien, Z. (2006). Gesture-based interface for home appliance control in smart home. In *Smart Homes and Beyond-ICOST2006 4th Int. Conference On Smart homes and health Telematics, C. Nugent and JC Augusto, Eds. IOS Press, Amsterdam*(pp. 23-30).

Döring, T., Kern, D., Marshall, P., Pfeiffer, M., Schöning, J., Gruhn, V., & Schmidt, A. (2011, May). Gestural interaction on the steering wheel: reducing the visual demand. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 483-492). ACM.

Dourish, P. (2004). What we talk about when we talk about context. *Personal and ubiquitous computing*, *8*(1), 19-30.

Dourish, P. (2004). *Where the action is: the foundations of embodied interaction*. MIT press.

Droeschel, D., Stuckler, J., Holz, D., & Behnke, S. (2011, May). Towards joint attention for a domestic service robot-person awareness and gesture recognition using time-of-flight cameras. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on* (pp. 1205-1210). IEEE.

Droeschel, D., Stuckler, J., Holz, D., & Behnke, S. (2011, May). Towards joint attention for a domestic service robot-person awareness and gesture recognition using time-of-flight cameras. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on* (pp. 1205-1210). IEEE.

Efron, D., & van Veen, S. (1972). *Gesture, race and culture*.

Eisenstein, J., & Davis, R. (2004, October). Visual and linguistic information in gesture classification. In *Proceedings of the 6th international conference on Multimodal interfaces* (pp. 113-120). ACM.

Ekman, P., & Friesen, W. V. (1972). Hand movements. *Journal of communication*, *22*(4), 353-374.

Endres, C., Schwartz, T., & Müller, C. A. (2011, February). Geremin: 2D microgestures for drivers based on electric field sensing. In *Proceedings of the 16th international conference on Intelligent user interfaces* (pp. 327-330). ACM.

Faltemier, T., Bowyer, K., & Flynn, P. (2006, June). 3D face recognition with region committee voting. In *3D Data Processing, Visualization, and Transmission, Third International Symposium on* (pp. 318-325). IEEE.

Feldman, R. S., & Rimé, B. (Eds.). (1991). *Fundamentals of nonverbal behavior*. Cambridge University Press.

Fishkin, K. P., Moran, T. P., & Harrison, B. L. (1999). Embodied user interfaces: Towards invisible user interfaces. In *Engineering for Human-Computer Interaction* (pp. 1-18). Springer US.

Fleer, D., & Leichsenring, C. (2012, October). MISO: a context-sensitive multimodal interface for smart objects based on hand gestures and finger snaps. In *Adjunct proceedings of the 25th annual ACM symposium on User interface software and technology* (pp. 93-94). ACM.

Fossati, A., Dimitrijevic, M., Lepetit, V., & Fua, P. (2010). From canonical poses to 3D motion capture using a single camera. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, *32*(7), 1165-1181.

Freedman, N. (1972). The analysis of movement behavior during the clinical interview. *Studies in dyadic communication*, 153-175.

Frisch, M., Heydekorn, J., & Dachselt, R. (2010). Diagram editing on interactive displays using multi-touch and pen gestures. In *Diagrammatic Representation and Inference* (pp. 182-196). Springer Berlin Heidelberg.

Gang, G., Min, C. H., & Kim, T. S. (2012, January). Design of bio-signal based physical activity monitoring system. In *Biomedical and Health Informatics (BHI), 2012 IEEE-EMBS International Conference on* (pp. 144-147). IEEE.

Gazendam, M. G., & Hof, A. L. (2007). Averaged EMG profiles in jogging and running at different speeds. *Gait & posture*, *25*(4), 604-614.

Gong, Y. (1995). Speech recognition in noisy environments: A survey. *Speech communication*, *16*(3), 261-291.

González, I. E., Wobbrock, J. O., Chau, D. H., Faulring, A., & Myers, B. A. (2007, May). Eyes on the road, hands on the wheel: thumb-based interaction techniques for input on steering wheels. In *Proceedings of Graphics Interface 2007* (pp. 95-102). ACM.

Greenberg, S., Marquardt, N., Ballendat, T., Diaz-Marino, R., & Wang, M. (2011). Proxemic interactions: the new ubicomp?. *interactions*, *18*(1), 42-50.

Gunes, H., & Piccardi, M. (2007). Bi-modal emotion recognition from expressive face and body gestures. *Journal of Network and Computer Applications*, *30*(4), 1334-1345.

Guo, C., & Sharlin, E. (2008, April). Exploring the use of tangible user interfaces for human-robot interaction: a comparative study. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 121-130). ACM.

Hall, E. T. (1963). A system for the notation of proxemic behavior1. *American anthropologist*, *65*(5), 1003-1026.

Harrison, B. L., Fishkin, K. P., Gujar, A., Mochon, C., & Want, R. (1998, January). Squeeze me, hold me, tilt me! An exploration of manipulative user interfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 17-24). ACM Press/Addison-Wesley Publishing Co..

Harrison, C., Benko, H., & Wilson, A. D. (2011, October). OmniTouch: wearable multitouch interaction everywhere. In *Proceedings of the 24th annual ACM symposium on User interface software and technology* (pp. 441-450). ACM.

Harrison, C., Tan, D., & Morris, D. (2010, April). Skinput: appropriating the body as an input surface. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 453-462). ACM.

Hendon, D. W., Hendon, R. A., & Herbig, P. A. (1996). *Cross-cultural business negotiations*. Greenwood Publishing Group.

Henricksen, K., & Indulska, J. (2004, March). A software engineering framework for context-aware pervasive computing. In *Pervasive Computing and Communications, 2004. PerCom 2004. Proceedings of the Second IEEE Annual Conference on* (pp. 77-86). IEEE.

Henze, N., Löcken, A., Boll, S., Hesselmann, T., & Pielot, M. (2010, December). Free-hand gestures for music playback: deriving gestures with a user-centred process. In *Proceedings of the 9th international conference on Mobile and Ubiquitous Multimedia* (p. 16). ACM.

Herda, L., Fua, P., Plankers, R., Boulic, R., & Thalmann, D. (2000). Skeleton-based motion capture for robust reconstruction of human motion. In *Computer Animation 2000. Proceedings* (pp. 77-83). IEEE.

Hermens, H. J., Freriks, B., Disselhorst-Klug, C., & Rau, G. (2000). Development of recommendations for SEMG sensors and sensor placement procedures. *Journal of electromyography and Kinesiology*, *10*(5), 361-374.

Hermens, H. J., Freriks, B., Merletti, R., Stegeman, D., Blok, J., Rau, G., ... & Hägg, G. (1999). European recommendations for surface electromyography.*Roessingh Research and Development, Enschede*.

Hinckley, K., Pausch, R., Proffitt, D., & Kassell, N. F. (1998). Two-handed virtual manipulation. *ACM Transactions on Computer-Human Interaction (TOCHI)*, *5*(3), 260-302.

Hinckley, K., Yatani, K., Pahud, M., Coddington, N., Rodenhouse, J., Wilson, A., ... & Buxton, B. (2010, October). Pen+ touch= new tools. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology*(pp. 27-36). ACM.

Hirschi, G. (2000). *Mudras: yoga in your hands*. Weiser Books.

Hofer, R. L., & Kunz, A. (2010, June). Digisketch: taming anoto technology on LCDs. In *Proceedings of the 2nd ACM SIGCHI symposium on Engineering interactive computing systems* (pp. 103-108). ACM.

Holte, M. B., Moeslund, T. B., & Fihl, P. (2010). View-invariant gesture recognition using 3D optical flow and harmonic motion context. *Computer Vision and Image Understanding*, *114*(12), 1353-1361.

Hong, J. Y., Suh, E. H., & Kim, S. J. (2009). Context-aware systems: A literature review and classification. *Expert Systems with Applications*, *36*(4), 8509-8522.

Hynes, G., Reynolds, V., & Hauswirth, M. (2009). A context lifecycle for web-based context management services. In *Smart Sensing and Context* (pp. 51-65). Springer Berlin Heidelberg.

Ishii, K., Zhao, S., Inami, M., Igarashi, T., & Imai, M. (2009). Designing laser gesture interface for robot control. In *Human-Computer Interaction–INTERACT 2009* (pp. 479-492). Springer Berlin Heidelberg.

Jacobson, E. (1931). Electrical measurements of neuromuscular states during mental activities. *American Journal of Physiology*, *96*, 115-121.

Jaimes, A., & Sebe, N. (2007). Multimodal human–computer interaction: A survey. *Computer vision and image understanding*, *108*(1), 116-134.

Jain, J., Lund, A., & Wixon, D. (2011, May). The future of natural user interfaces. In *CHI'11 Extended Abstracts on Human Factors in Computing Systems* (pp. 211-214). ACM.

Jiang, Y., Tian, F., Zhang, X., Liu, W., Dai, G., & Wang, H. (2012, February). Unistroke gestures on multi-touch interaction: Supporting flexible touches with key stroke extraction. In *Proceedings of the 2012 ACM international conference on Intelligent User Interfaces* (pp. 85-88). ACM.

Jing, L., Zhou, Y., Cheng, Z., & Huang, T. (2012). Magic ring: A finger-worn device for multiple appliances control using static finger gestures. *Sensors*,*12*(5), 5775-5790.

Jojic, N., Brumitt, B., Meyers, B., Harris, S., & Huang, T. (2000). Detection and estimation of pointing gestures in dense disparity maps. In *Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on* (pp. 468-475). IEEE.

Jurman, G., Riccadonna, S., & Furlanello, C. (2012). A comparison of MCC and CEN error measures in multi-class prediction. *PloS one*, *7*(8), e41882.

Kahn, R. E., Swain, M. J., Prokopowicz, P. N., & Firby, R. J. (1996, June). Gesture recognition using the perseus architecture. In *Computer Vision and Pattern Recognition, 1996. Proceedings CVPR'96, 1996 IEEE Computer Society Conference on* (pp. 734-741). IEEE.

Karam, M. (2006). *PhD Thesis: A framework for research and design of gesture-based human-computer interactions* (Doctoral dissertation, University of Southampton).

Kela, J., Korpipää, P., Mäntyjärvi, J., Kallio, S., Savino, G., Jozzo, L., & Marca, D. (2006). Accelerometer-based gesture control for a design environment.*Personal and Ubiquitous Computing*, *10*(5), 285-299.

Kendon, A. (1986). Some reasons for studying gesture. *Semiotica*, *62*(1-2), 3-28.

Kendon, A. (1995). Gestures as illocutionary and discourse structure markers in Southern Italian conversation. *Journal of pragmatics*, *23*(3), 247-279.

Kendon, A. (2000). Language and gesture: Unity or duality. *Language and gesture*, 47-63.

Kettebekov, S. (2004, October). Exploiting prosodic structuring of coverbal gesticulation. In *Proceedings of the 6th international conference on Multimodal interfaces* (pp. 105-112). ACM.

Khoshelham, K., & Elberink, S. O. (2012). Accuracy and resolution of kinect depth data for indoor mapping applications. *Sensors*, *12*(2), 1437-1454.

Kim, H. J., Jeong, K. H., Kim, S. K., & Han, T. D. (2011, December). Ambient wall: Smart wall display interface which can be controlled by simple gesture for smart home. In *SIGGRAPH Asia 2011 Sketches* (p. 1). ACM.

Kittler, J., & Alkoot, F. M. (2003). Sum versus vote fusion in multiple classifier systems. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*,*25*(1), 110-115.

Kittler, J., Hatef, M., Duin, R. P., & Matas, J. (1998). On combining classifiers.*Pattern Analysis and Machine Intelligence, IEEE Transactions on*, *20*(3), 226-239.

Kivimaki, T., Vuorela, T., Valtonen, M., & Vanhala, J. (2013, July). Gesture Control System for Smart Environments. In *Intelligent Environments (IE), 2013 9th International Conference on* (pp. 232-235). IEEE.

Klein, G., & Murray, D. (2007, November). Parallel tracking and mapping for small AR workspaces. In *Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on* (pp. 225-234). IEEE.

Klima, E. S. (1979). *The signs of language*. Harvard University Press.

Kocaballi, A. B. (2010, August). Wearable environments: reconfiguring human-machine-environment relations. In *Proceedings of the 28th Annual European Conference on Cognitive Ergonomics* (pp. 315-318). ACM.

Kofod-Petersen, A., & Cassens, J. (2006). Using activity theory to model context awareness. In *Modeling and Retrieval of Context* (pp. 1-17). Springer Berlin Heidelberg.

Krahnstoever, N., Kettebekov, S., Yeasin, M., & Sharma, R. (2002, October). A real-time framework for natural multimodal interaction with large screen displays. In *Proceedings of the 4th IEEE International Conference on Multimodal Interfaces* (p. 349). IEEE Computer Society.

Kühnel, C., Westermann, T., Hemmert, F., Kratz, S., Müller, A., & Möller, S. (2011). I'm home: Defining and evaluating a gesture set for smart-home control.*International Journal of Human-Computer Studies*, *69*(11), 693-704.

Kuncheva, L. I., Bezdek, J. C., & Duin, R. P. (2001). Decision templates for multiple classifier fusion: an experimental comparison. *Pattern Recognition*,*34*(2), 299-314.

LaFrance, M., & Mayo, C. (1978). Cultural aspects of nonverbal communication.*International Journal of Intercultural Relations*, *2*(1), 71-89.

LaViola, J. J. (2013). 3D Gestural Interaction: The State of the Field.*International Scholarly Research Notices*, *2013*.

Lee, M., Billinghurst, M., Baek, W., Green, R., & Woo, W. (2013). A usability study of multimodal input in an augmented reality environment. *Virtual Reality*,*17*(4), 293-305.

Lenman, S., Bretzner, L., & Thuresson, B. (2002, October). Using marking menus to develop command sets for computer vision based hand gesture interfaces. In *Proceedings of the second Nordic conference on Human-computer interaction* (pp. 239-242). ACM.

Lewis, J. R., & Sauro, J. (2009). The factor structure of the system usability scale. In *Human Centered Design* (pp. 94-103). Springer Berlin Heidelberg.

Lewis, M. Paul; Simons, Gary F.; Fennig, Charles D., eds. (2013). "Deaf sign language". *Ethnologue: Languages of the World* (17th ed.). SIL International.

Liang, J., & Green, M. (1994). JDCAD: A highly interactive 3D modeling system. *Computers & graphics*, *18*(4), 499-506.

Liddell, S. K. (2003). *Grammar, gesture, and meaning in American Sign Language*. Cambridge University Press.

Linz, T., Gourmelon, L., & Langereis, G. (2007, January). Contactless EMG sensors embroidered onto textile. In *4th International Workshop on Wearable and Implantable Body Sensor Networks (BSN 2007)* (pp. 29-34). Springer Berlin Heidelberg.

Malik, S., & Laszlo, J. (2004, October). Visual touchpad: a two-handed gestural input device. In *Proceedings of the 6th international conference on Multimodal interfaces* (pp. 289-296). ACM.

Mann, S., Nolan, J., & Wellman, B. (2002). Sousveillance: Inventing and Using Wearable Computing Devices for Data Collection in Surveillance Environments.*Surveillance & Society*, *1*(3), 331-355.

Matthews, B. W. (1975). Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochimica et Biophysica Acta (BBA)-Protein Structure*, *405*(2), 442-451.

McNeill, D. (1985). So you think gestures are nonverbal?. *Psychological review*,*92*(3), 350.

McNeill, D. (1987). *Psycholinguistics: A new approach*. Harper & Row Publishers.

McNeill, D., & Levy, E. (1980). Conceptual representations in language activity and gesture.

Miller, G. A. (1956). The magical number seven, plus or minus two: some limits on our capacity for processing information. *Psychological review*, *63*(2), 81.

Minsky, M. R. (1984, January). Manipulating simulated objects with real-world gestures using a force and position sensitive screen. In *ACM SIGGRAPH Computer Graphics* (Vol. 18, No. 3, pp. 195-203). ACM.

Mistry, P., & Maes, P. (2009, December). SixthSense: a wearable gestural interface. In *ACM SIGGRAPH ASIA 2009 Sketches* (p. 11). ACM.

Mistry, P., Maes, P., & Chang, L. (2009, April). WUW-wear Ur world: a wearable gestural interface. In *CHI'09 extended abstracts on Human factors in computing systems* (pp. 4111-4116). ACM.

Mitra, S., & Acharya, T. (2007). Gesture recognition: A survey. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*,*37*(3), 311-324.

Moran, T. P., & Dourish, P. (2001). Introduction to this special issue on context-aware computing. *Human–Computer Interaction*, *16*(2-4), 87-95.

Murer, M., Wilfinger, D., Meschtscherjakov, A., Osswald, S., & Tscheligi, M. (2012, October). Exploring the back of the steering wheel: Text input with hands on the wheel and eyes on the road. In *Proceedings of the 4th International Conference on Automotive User Interfaces and Interactive Vehicular Applications* (pp. 117-120). ACM.

Nacenta, M. A., Kamber, Y., Qiang, Y., & Kristensson, P. O. (2013, April). Memorability of pre-designed and user-defined gesture sets. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 1099-1108). ACM.

Neßelrath, R., Lu, C., Schulz, C. H., Frey, J., & Alexandersson, J. (2011). A Gesture Based System for Context–Sensitive Interaction with Smart Homes. In*Ambient Assisted Living* (pp. 209-219). Springer Berlin Heidelberg.

Nickel, K., & Stiefelhagen, R. (2007). Visual recognition of pointing gestures for human–robot interaction. *Image and Vision Computing*, *25*(12), 1875-1884.

Niedermaier, B., Durach, S., Eckstein, L., & Keinath, A. (2009). The new BMW iDrive–applied processes and methods to assure high usability. In *Digital Human Modeling* (pp. 443-452). Springer Berlin Heidelberg.

Nielsen, J. (2003). Usability 101: Introduction to usability. http://www.nngroup.com/articles/usability-101-introduction-to-usability. Site accessed on 26 June 2014

Nielsen, M., Störring, M., Moeslund, T. B., & Granum, E. (2004). A procedure for developing intuitive and ergonomic gesture interfaces for HCI. In *Gesture-Based Communication in Human-Computer Interaction* (pp. 409-420). Springer Berlin Heidelberg.

Norman, D. A. (2010). Natural user interfaces are not natural. *interactions*,*17*(3), 6-10.

Norman, D. A., & Nielsen, J. (2010). Gestural interfaces: a step backward in usability. *interactions*, *17*(5), 46-49.

Oh, K., Jeong, Y. S., Kim, S. S., & Choi, H. J. (2011, February). Gesture recognition application with Parametric Hidden Markov Model for activity-based personalized service in APRiME. In *Cognitive Methods in Situation Awareness and Decision Support (CogSIMA), 2011 IEEE First International Multi-Disciplinary Conference on* (pp. 189-193). IEEE.

Oh, U., & Findlater, L. (2013, April). The challenges and potential of end-user gesture customization. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 1129-1138). ACM.

Paulson, B., Cummings, D., & Hammond, T. (2011). Object interaction detection using hand posture cues in an office setting. *International journal of human-computer studies*, *69*(1), 19-29.

Peng, B., Qian, G., & Rajko, S. (2009, August). View-invariant full-body gesture recognition via multilinear analysis of voxel data. In *Distributed Smart Cameras, 2009. ICDSC 2009. Third ACM/IEEE International Conference on* (pp. 1-8). IEEE.

Pentland, A. (1998, August). Smart rooms, smart clothes. In *Pattern Recognition, 1998. Proceedings. Fourteenth International Conference on* (Vol. 2, pp. 949-953). IEEE.

Pfleging, B., Schneegass, S., & Schmidt, A. (2012, October). Multimodal interaction in the car: combining speech and gestures on the steering wheel. In*Proceedings of the 4th International Conference on Automotive User Interfaces and Interactive Vehicular Applications* (pp. 155-162). ACM.

Pika, S., Nicoladis, E., & Marentette, P. (2009). How to Order a Beer Cultural Differences in the Use of Conventional Gestures for Numbers. *Journal of Cross-Cultural Psychology*, *40*(1), 70-80.

Price, A. W. (2008). *Contextuality in practical reason*. Oxford University Press.

Quek, F.(2004). Gesture Recognition. *Encyclopedia of Human-Computer Interaction*, Vol. 1, pp. 288-292, Berkshire Publishing Group.

Quek, F., McNeill, D., Bryll, R., Duncan, S., Ma, X. F., Kirbas, C., ... & Ansari, R. (2002). Multimodal human discourse: gesture and speech. *ACM Transactions on Computer-Human Interaction (TOCHI)*, *9*(3), 171-193.

Rabiner, L. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, *77*(2), 257-286.

Raffa, G., Lee, J., Nachman, L., & Song, J. (2010, October). Don't slow me down: Bringing energy efficiency to continuous gesture recognition. In *Wearable Computers (ISWC), 2010 International Symposium on* (pp. 1-8). IEEE.

Ramey, A., González-Pacheco, V., & Salichs, M. A. (2011, March). Integration of a low-cost RGB-D sensor in a social robot for gesture recognition. In*Proceedings of the 6th international conference on Human-robot interaction* (pp. 229-230). ACM.

Refaeilzadeh, P., Tang, L., & Liu, H. (2009). Cross-validation. In *Encyclopedia of database systems* (pp. 532-538). Springer US.

Rekimoto, J. (1997, October). Pick-and-drop: a direct manipulation technique for multiple computer environments. In *Proceedings of the 10th annual ACM symposium on User interface software and technology* (pp. 31-39). ACM.

Rekimoto, J. (2002, April). SmartSkin: an infrastructure for freehand manipulation on interactive surfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 113-120). ACM.

Reuter, P., Riviere, G., Couture, N., Mahut, S., & Espinasse, L. (2010). ArcheoTUI—Driving virtual reassemblies with tangible 3D interaction. *Journal on Computing and Cultural Heritage (JOCCH)*, *3*(2), 4.

Rhodes, B. J., Minar, N., & Weaver, J. (1999, October). Wearable computing meets ubiquitous computing: Reaping the best of both worlds. In *Wearable Computers, 1999. Digest of Papers. The Third International Symposium on* (pp. 141-149). IEEE.

Riener, A. (2012). Gestural interaction in vehicular applications. *Computer*,*45*(4), 42-47.

Rimé, B. (1983). Nonverbal communication or nonverbal behavior? Towards a cognitive-motor theory of nonverbal behavior. *Current issues in European social psychology*, *1*, 85-141.

Roggen, D., Lukowicz, P., Ferscha, A., Millán, J. D. R., Tröster, G., & Chavarriaga, R. (2013). Opportunistic human activity and context recognition.*Computer-IEEE Computer Society-*, *46*(EPFL-ARTICLE-182084), 36-45.

Roh, M. C., Shin, H. K., Lee, S. W., & Lee, S. W. (2006, August). Volume motion template for view-invariant gesture recognition. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on* (Vol. 2, pp. 1229-1232). IEEE.

Rowley, J. E. (2007). The wisdom hierarchy: representations of the DIKW hierarchy. *Journal of Information Science*.

Rubine, D. (1992, June). Combining gestures and direct manipulation. In*Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 659-660). ACM.

Saponas, T. S., Tan, D. S., Morris, D., & Balakrishnan, R. (2008, April). Demonstrating the feasibility of using forearm electromyography for muscle-computer interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 515-524). ACM.

Sato, E., Sakurai, S., Nakajima, A., Yoshida, Y., & Yamguchi, T. (2007, August). Context-based interaction using pointing movements recognition for an intelligent home service robot. In *Robot and Human interactive Communication,*

*2007. RO-MAN 2007. The 16th IEEE International Symposium on* (pp. 854-859). IEEE.

Schilit, B., Adams, N., & Want, R. (1994, December). Context-aware computing applications. In *Mobile Computing Systems and Applications, 1994. WMCSA 1994. First Workshop on* (pp. 85-90). IEEE.

Schlömer, T., Poppinga, B., Henze, N., & Boll, S. (2008, February). Gesture recognition with a Wii controller. In *Proceedings of the 2nd international conference on Tangible and embedded interaction* (pp. 11-14). ACM.

Schuller, B., Wöllmer, M., Moosmayr, T., & Rigoll, G. (2009). Recognition of noisy speech: A comparative survey of robust model architecture and feature enhancement. *EURASIP Journal on Audio, Speech, and Music Processing*,*2009*, 5.

Selker, T., & Burleson, W. (2000). Context-aware design and interaction in computer systems. *IBM systems Journal*, *39*(3.4), 880-891.

Shafer, S. A., Brumitt, B., & Cadiz, J. J. (2001). Interaction issues in context-aware intelligent environments. *Human-Computer Interaction*, *16*(2), 363-378.

Shneiderman, B. (2000). The limits of speech recognition. *Communications of the ACM*, *43*(9), 63-65.

Skelton, I., & Cooper, J. (2004). You're not from around here, are you?. *Joint Force Quarterly*, *36*, 12-16.

Soechting, J. F., & Ross, B. (1984). Psychophysical determination of coordinate representation of human arm orientation. *Neuroscience*, *13*(2), 595-604.

Soechting, J. F., Buneo, C. A., Herrmann, U., & Flanders, M. (1995). Moving effortlessly in three dimensions: does Donders' law apply to arm movement?.*The Journal of Neuroscience*, *15*(9), 6271-6280.

Soechting, J. F., Lacquaniti, F., & Terzuolo, C. A. (1986). Coordination of arm movements in three-dimensional space. Sensorimotor mapping during drawing movement. *Neuroscience*, *17*(2), 295-311.

Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management*,*45*(4), 427-437.

Sommaruga, L., Formilli, T., & Rizzo, N. (2011, September). DomoML: an integrating devices framework for ambient intelligence solutions. In *Proceedings*

*of the 6th International Workshop on Enhanced Web Service Technologies* (pp. 9-15). ACM.

Starner, T. (2013). Project glass: An extension of the self. *Pervasive Computing, IEEE*, *12*(2), 14-16.

Starner, T., Auxier, J., Ashbrook, D., & Gandy, M. (2000, October). The gesture pendant: A self-illuminating, wearable, infrared computer vision system for home automation control and medical monitoring. In *Wearable computers, the fourth international symposium on* (pp. 87-94). IEEE.

Starner, T., Weaver, J., & Pentland, A. (1998). A wearable computer based american sign language recognizer. In *Assistive Technology and Artificial Intelligence* (pp. 84-96). Springer Berlin Heidelberg.

Starner, T., Weaver, J., & Pentland, A. (1998). Real-time american sign language recognition using desk and wearable computer based video. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, *20*(12), 1371-1375.

Stocklöw, C., & Wichert, R. (2012). Gesture Based Semantic Service Invocation for Human Environment Interaction. In *Ambient Intelligence* (pp. 304-311). Springer Berlin Heidelberg.

Striegnitz, K., Tepper, P., Lovett, A., & Cassell, J. (2005). Knowledge representation for generating locating gestures in route directions. *WS in Spatial Language and Dialogue*.

Studdert-Kennedy, M. (1987). The phoneme as a perceptuomotor structure. In A. Allport, D. G. MacKay, W. Prinz & E. Scheerer (eds.), Language Perception and Production : Relationships between Listening, Speaking, Reading and Writing, pp.67-84, London : Academic Press.

Surie, D., Jackel, F., Janlert, L., & Pederson, T. (2010, July). Situative space tracking within smart environments. In *Intelligent Environments (IE), 2010 Sixth International Conference on* (pp. 152-157). IEEE.

Surie, D., Pederson, T., Lagriffoul, F., Janlert, L. E., & Sjölie, D. (2007). *Activity recognition using an egocentric perspective of everyday objects* (pp. 246-257). Springer Berlin Heidelberg.

Swiss Federation. (2011). Swiss average utilization of transport means in 2010. http://www.bfs.admin.ch/bfs/portal/fr/index/themen/11/07/01/01/unterwegszeiten/01.html. Site accessed: 30 March 2013.

Tashev, I., Seltzer, M., Ju, Y. C., Wang, Y. Y., & Acero, A. (2009). Commute UX: Voice enabled in-car infotainment system. In *Mobile HCI* (Vol. 9).

Thalmann, D., Shen, J., & Chauvineau, E. (1996, June). Fast realistic human body deformations for animation and vr applications. In *Computer Graphics International, 1996. Proceedings* (pp. 166-174). IEEE.

The Telegraph. (2011). Britons spend more time driving than socialising. http://www.telegraph.co.uk/motoring/news/8287098/Britons-spend-more-time-driving-than-socialising.html. Site accessed: 26 June 2014.

Turk, M. (2005). Multimodal human-computer interaction. In *Real-time vision for human-computer interaction* (pp. 269-283). Springer US.

Underkoffler, J., & Ishii, H. (1999, May). Urp: a luminous-tangible workbench for urban planning and design. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems* (pp. 386-393). ACM.

Valli, A. (2008). The design of natural interaction. *Multimedia Tools and Applications*, *38*(3), 295-305.

Van den Bergh, M., & Van Gool, L. (2011, January). Combining RGB and ToF cameras for real-time 3D hand gesture interaction. In *Applications of Computer Vision (WACV), 2011 IEEE Workshop on* (pp. 66-72). IEEE.

van den Hoven, E., & Mazalek, A. (2011). Grasping gestures: Gesturing with physical artifacts. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, *25*(03), 255-271.

van den Hoven, E., van de Garde-Perik, E., Offermans, S., van Boerdonk, K., & Lenssen, K. M. H. (2013). Moving tangible interaction systems to the next level.*Computer*, *46*(8), 0070-76.

van Ingen, S. G. (1979). Some fundamental aspects of the biomechanics of overground versus treadmill locomotion. *Medicine and Science in Sports and Exercise*, *12*(4), 257-261.

Victor, T., & Dalzell, T. (2007). *The concise new Partridge dictionary of slang and unconventional English*. Routledge.

Villegas, N. M., & Müller, H. A. (2010). Managing dynamic context to optimize smart interactions and services. In *The smart internet* (pp. 289-318). Springer Berlin Heidelberg.

Vogel, D., & Balakrishnan, R. (2004, October). Interactive public ambient displays: transitioning from implicit to explicit, public to personal, interaction with multiple users. In *Proceedings of the 17th annual ACM symposium on User interface software and technology* (pp. 137-146). ACM.

Volterra, V., Caselli, M. C., Capirci, O., & Pizzuto, E. (2005). Gesture and the emergence and development of language. *Beyond nature-nurture: Essays in honor of Elizabeth Bates*, 3-40.

Wachs, J. P., Kölsch, M., Stern, H., & Edan, Y. (2011). Vision-based hand-gesture applications. *Communications of the ACM*, *54*(2), 60-71.

Weibel, P. (1994). Kontext Kunst: The Art of the 90's.

Weiser, M. (1991). The computer for the 21st century. *Scientific american*,*265*(3), 94-104.

Weiser, M., & Brown, J. S. (1997). The coming age of calm technology. In *Beyond calculation* (pp. 75-85). Springer New York.

Weiser, M., Gold, R., & Brown, J. S. (1999). The origins of ubiquitous computing research at PARC in the late 1980s. *IBM systems journal*, *38*(4), 693-696.

Welch, L. R. (2003). Hidden Markov models and the Baum-Welch algorithm.*IEEE Information Theory Society Newsletter*, *53*(4), 10-13.

Wenhui, W., Xiang, C., Kongqiao, W., Xu, Z., & Jihai, Y. (2009, September). Dynamic gesture recognition based on multiple sensors fusion technology. In*Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE* (pp. 7014-7017). IEEE.

Wexelblat, A. (1995). An approach to natural gesture in virtual environments.*ACM Transactions on Computer-Human Interaction (TOCHI)*, *2*(3), 179-200.

Wexelblat, A. (1998). Research challenges in gesture: Open issues and unsolved problems. In *Gesture and sign language in human-computer interaction* (pp. 1-11). Springer Berlin Heidelberg.

Wexelblat, A. (1998). Research challenges in gesture: Open issues and unsolved problems. In *Gesture and sign language in human-computer interaction* (pp. 1-11). Springer Berlin Heidelberg.

Wickens, C. D. (1992). *Engineering psychology and human performance* . HarperCollins Publishers.

Widjaja, I., & Balbo, S. (2005, November). Spheres of role in context-awareness. In *Proceedings of the 17th Australia conference on Computer-Human Interaction: Citizens Online: Considerations for Today and the Future*(pp. 1-4). Computer-Human Interaction Special Interest Group (CHISIG) of Australia.

Wigdor, D., & Wixon, D. (2011). *Brave NUI world: designing natural user interfaces for touch and gesture*. Elsevier.

Wilson, A. D. (2004, October). TouchLight: an imaging touch screen and display for gesture-based interaction. In *Proceedings of the 6th international conference on Multimodal interfaces* (pp. 69-76). ACM.

Winter, D. A., & Yack, H. J. (1987). EMG profiles during normal human walking: stride-to-stride and inter-subject variability. *Electroencephalography and clinical neurophysiology*, *67*(5), 402-411.

Wobbrock, J. O., Morris, M. R., & Wilson, A. D. (2009, April). User-defined gestures for surface computing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 1083-1092). ACM.

Wolf, K., Naumann, A., Rohs, M., & Müller, J. (2011). A taxonomy of microinteractions: Defining microgestures based on ergonomic and scenario-dependent requirements. In *Human-Computer Interaction–INTERACT 2011* (pp. 559-575). Springer Berlin Heidelberg.

Wu, J., Pan, G., Zhang, D., Li, S., & Wu, Z. (2010, September). MagicPhone: pointing & interacting. In *Proceedings of the 12th ACM international conference adjunct papers on Ubiquitous computing-Adjunct* (pp. 451-452). ACM.

Wu, M., & Balakrishnan, R. (2003, November). Multi-finger and whole hand gestural interaction techniques for multi-user tabletop displays. In *Proceedings of the 16th annual ACM symposium on User interface software and technology*(pp. 193-202). ACM.

Wu, Y., & Huang, T. S. (2000). View-independent recognition of hand postures. In *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on* (Vol. 2, pp. 88-94). IEEE.

Yin, X., & Zhu, X. (2006, May). Hand posture recognition in gesture-based human-robot interaction. In *Industrial Electronics and Applications, 2006 1ST IEEE Conference on* (pp. 1-6). IEEE.

Youngblood, G. M., Cook, D. J., & Holder, L. B. (2005). Managing adaptive versatile environments. *Pervasive and Mobile Computing*, *1*(4), 373-403.

Yuan, R., Cheng, J., Li, P., Chen, G., Xie, C., & Xie, Q. (2010, July). View invariant hand gesture recognition using 3D trajectory. In *Intelligent Control and Automation (WCICA), 2010 8th World Congress on* (pp. 6315-6320). IEEE.

Zhao, S., & Balakrishnan, R. (2004, October). Simple vs. compound mark hierarchical marking menus. In *Proceedings of the 17th annual ACM symposium on User interface software and technology* (pp. 33-42). ACM.

Zimmermann, A., Lorenz, A., & Oppermann, R. (2007). An operational definition of context. In *Modeling and using context* (pp. 558-571). Springer Berlin Heidelberg.